

Estudio Empírico del Estado Actual de la Estimación de Software en Pymes de Colombia

Iván Darío Páez Anaya

Mayo 2012



UNIVERSIDAD EAFIT

PROYECTO DE GRADO PARA OPTAR AL TÍTULO DE MAESTRIA

Estudio Empírico del Estado Actual de la Estimación de Software en Pymes de Colombia

Author:

Iván Darío PÁEZ ANAYA

Supervisores:

Dr. Raquel ANAYA

Dr. Guilherme H. TRAVASSOS

Departamento de Informática y Sistemas

Escuela de Ingeniería

Universidad EAFIT

Mayo 2012

Agradecimientos

A Dios, quien siempre ha estado conmigo, me ha dado fuerza y ha guiado mis pasos en todos los caminos que he transitado.

Agradezco mis padres y a mis hermanos, quienes siempre han estado conmigo, me han apoyado desde el principio y siempre han creído en mí.

Agradezco a mi novia, por su comprensión y que incondicionalmente siempre me ha motivado a continuar adelante.

Agradezco a mi asesora, Raquel Anaya quien han compartido su conocimiento y me han brindado las herramientas necesarias para poder culminar exitosamente este estudio.

Agradezco a mi co-asesor, Guilherme H. Travassos por sus enseñanzas y directrices y por mostrarme el camino hacia los estudios experimentales que serán muy valiosos en mi carrera como investigador.

Agradezco a la Universidad EAFIT y sus profesores los cuales han contribuido en mi formación, no solo como profesional pero tan bien a nivel personal.

Agradezco a mis compañeros de trabajo, colegas y compañeros de universidad, los cuales han discutido y aportado ideas para este trabajo.

Agradezco a la Red Colombiana de Calidad para el Desarrollo en Tecnologías de Información y Comunicación RCC-TIC, participante clave para realizar el estudio experimental en las Pymes colombianas.

Agradezco a la Red LACCIR por el apoyo brindado para realizar una estancia corta de investigación en la Universidad Federal de Rio de Janeiro, UFRJ.

Agradezco a los compañeros integrantes del Grupo de Investigación en Ingeniería de Software Experimental PESCE-ESE /COPPE/UFRJ, por su acogida e ideas compartidas durante mi estadía en este lugar.

Resumen

Este trabajo de maestría consiste en un estudio experimental enfocado en analizar el estado actual de las prácticas de estimación de software de las Pymes colombianas, que realizaron un primer ciclo de mejora utilizando CMMI-DEV v1.2 [13] como modelo referente. Parte de este documento contiene el protocolo del estudio experimental, junto con las características de este estudio, lo cual facilita la comprensión de los objetivos, preguntas de investigación y resultados encontrados al final del estudio.

El objetivo principal de este trabajo de maestría fue caracterizar los procesos de estimación de proyectos de software de las Pymes desde dos perspectivas: primero desde la perspectiva del proceso, se analiza el nivel de cubrimiento de buenas prácticas de estimación descritas en el modelo CMMI-DEV v1.2 y segundo desde la perspectiva de los métodos de estimación se busca caracterizar cuales son los métodos más usados en la industria. Para recopilar la información se diseñó y aplicó una encuesta a las Pymes que estuvieron vinculadas a un programa masivo de mejora llevado a cabo durante el periodo 2008-2010 en Colombia. [48].

Así mismo, se analizan la precisión de la estimación del esfuerzo (horas hombre) y la duración (semanas) de proyectos de software. Tal como lo plantea [19], es importante analizar estas dos variables por separado ya que algunas empresas ponen mayor empeño en terminar los proyectos a tiempo y para lograr esto sobrecargan al personal para cumplir con estas metas, esto puede generar un falso sentido de éxito, ya que el éxito de un proyecto depende de entregar toda la funcionalidad a tiempo y de acuerdo al esfuerzo estimado inicialmente.

Desde la perspectiva de los métodos de estimación se desea indagar el grado de aplicación en la industria de las diversas aproximaciones para estimar el esfuerzo de desarrollo: métodos basados en juicio de expertos tales como : Delphi, Delphi de Banda Ancha [55], [68], EDT Estructura de Descomposición del Trabajo o WBS [20] y Planing Poker [43], [50] , métodos basados en modelos formales que pueden ser de tipo paramétrico por ejemplo: COCOMO II [7] , [13], [59], modelos formales basados en tamaño Puntos de Casos de Uso [3], [3], [12] y métodos que realizan una combinación de varios, ejemplo [5], [44].

Los resultados obtenidos permiten afirmar que las compañías de software analizadas realizan la estimación de los proyectos de manera intuitiva y su precisión en la estimación del esfuerzo y la calendarización es baja. Por lo tanto hay muchas oportunidades de mejora en este frente de trabajo.

Palabras claves: Estimación de Software, Métodos de Estimación, CMMI-DEV v1.2, Ingeniería de Software Experimental, Estudio Experimental, Estudio Empírico.

Índice

Índice	III
Índice de Figuras	VI
Índice de Tablas	VII
Lista de Abreviaturas	VIII
1. Introducción	1
1.1. Motivación	1
1.2. Objetivos	3
1.3. Metodología de Trabajo	3
1.4. Organización	4
1.5. Síntesis de los resultados alcanzados	5
2. Conceptos y Trabajos Relacionados	6
2.1. La naturaleza de la estimación de proyectos de software.	6
2.1.1. Relación entre la estimación y el plan del proyecto.	7
2.1.2. Un buen nivel de confianza en la estimación representa una ventaja competitiva para la organización.	7
2.1.3. La estimación es tanto arte como ciencia	8
2.1.4. Influencia de la dinámica del proyecto en las estimaciones	8
2.1.5. La importancia de los datos históricos como base de estimación.	10
2.2. Métodos de estimación de software	11
2.2.1. Métodos de medición de tamaño funcional	11
2.2.2. Métodos de estimación basados en parámetros	16
2.2.3. Métodos de estimación basados en la experiencia o heurísticos	20
2.3. Conceptos básicos de la Ingeniería de Software Experimental	24
2.3.1. Importancia y retos de la experimentación en Ingeniería de Software	24
2.3.2. Dificultades de la experimentación.	25

2.3.3.	Elementos de la experimentación	25
2.3.4.	Definición del objetivo del experimento	27
2.3.5.	Proceso del estudio experimental	28
2.4.	Trabajos experimentales relacionados a la estimación de software	28
3.	Definición del Estudio Experimental en Estimación de Software	32
3.1.	Caracterización del estudio	32
3.1.1.	Tipo o de estudio Encuesta.	32
3.1.2.	Dominio.	32
3.1.3.	Idioma.	33
3.1.4.	Participantes	33
3.1.5.	Número de repeticiones estimadas.	33
3.2.	Definición del estudio experimental	33
3.2.1.	Objetivo General	33
3.2.2.	Objetivos Específicos	34
3.2.3.	Foco de calidad.	34
3.3.	Perspectiva de observación	38
3.4.	Selección del contexto	38
3.5.	Preguntas de investigación y métricas asociadas a estas	39
3.6.	Planeación	42
3.6.1.	Formulación de hipótesis	42
3.6.2.	Selección de variables	44
3.6.3.	Selección de personal	45
3.7.	Diseño del experimento	46
3.7.1.	Instrumentación	47
3.7.2.	Validez de resultados	51
4.	Análisis de los Datos Recolectados	53
4.1.	Información general	53
4.2.	Respuestas a las preguntas de investigación	54
4.2.1.	RQ1. ¿Cuáles son las prácticas de estimación más aplicadas en las empresas?	58
4.2.2.	RQ2. ¿Cuáles es el nivel de aplicación de las prácticas consideradas útiles para la organización?	59
4.2.3.	RQ3. ¿Cuál es la precisión de estimación del esfuerzo en los proyectos de software?	60
4.2.4.	RQ4. ¿Cuál es la precisión de estimación de la calendarización en los proyectos de software?	61
4.2.5.	RQ5. ¿Cuáles son las oportunidades de mejora y factores de éxito que perciben las compañías en el proceso de estimación?	63

4.2.6.	RQ6. ¿Cuál es el nivel de aplicación en la organización de los métodos de estimación reconocidos en la literatura?	64
4.2.7.	RQ7. ¿De qué manera se aplican en la organización los métodos de estimación?	66
4.2.8.	RQ8. ¿Cuál es el nivel de satisfacción de los representantes de la organización con respecto a la estimación?	67
4.3.	Análisis de hipótesis	68
4.4.	Limitaciones	68
5.	Conclusiones y Trabajos Futuros	70
5.1.	Conclusiones	70
5.2.	Trabajos futuros	71
5.3.	Publicaciones relacionadas.	72
	Bibliografía	73

Índice de Figuras

1.1. Metodología de Trabajo, adaptado de [64].	3
2.1. Influencia de la dinámica del proyecto en las estimaciones. Adaptado de [41].	9
2.2. Evolución de los métodos estándares. Adaptado de [15].	16
2.3. COCOMO II - Constructive Cost Model. Tomado de [14].	19
2.4. Los conceptos de un experimento. Tomado de [67], [70].	27
2.5. Proceso del Estudio Experimental. Tomado de [64], [65]	28
3.1. Modelo conceptual de la encuesta. Adaptado de [67].	35
4.1. Distribución geográfica de las empresas que participaron en el estudio. . . .	54
4.2. Distribución según número de empleados.	55
4.3. Distribución según Modelos/Estándares adoptados.	55
4.4. Frecuencia encontrada de las prácticas de estimación CMMI-DEV v1.2. . . .	58
4.5. Distribución del porcentaje de precisión de la estimación de esfuerzo (PPEE)	61
4.6. Distribución del porcentaje de precisión de la estimación de la duración (PPED)	62
4.7. Aplicación de los métodos de estimación	65
4.8. Nivel de aplicación de los métodos de estimación	66
4.9. Nivel de satisfacción con el proceso de estimación.	67

Índice de Tablas

1.1. Distribución de las empresas participantes en el proyecto.	2
2.1. Matriz de valores de FP sin ajustar. Tomado de [46].	15
2.2. Características generales del sistema para FPA. Tomado de [46].	15
2.3. Estimadores de costo (Cost Drivers) COCOMO II. Tomado de [60]	18
2.4. Factores de escala (scale factors) COCOMO II. Tomado de [60]	18
2.5. Síntesis de Trabajos Relacionados	30
2.6. Comparación de la precisión de la estimación. Adaptado de [19].	31
3.1. Prácticas específicas de estimación sugeridas por CMMI-DEV v1.2	36
3.2. Prácticas genéricas de estimación sugeridas por CMMI-DEV v1.2	37
3.3. Métodos de estimación seleccionados para el estudio. Adaptado de [45]. . .	38
3.4. Nivel de conocimiento de los métodos de estimación.	50
4.1. Distribución de las empresas participantes en el proyecto.	57

Lista de Abreviaturas

CMMI-DEV	Capability Maturity Model Integration for Development
COCOMO	Constructive Cost Model
COPPE/UFRJ	Instituto Alberto Luis Coimbra de Pós-Graduação e Pesquisa de Engenharia da Universidade Federal do Rio de Janeiro
COSMIC	Common Software Measurement International Consortium
CSBSG	Chinese Software Benchmarking Standards Group
DPE	Duración Promedio Estimada
DPR	Duración Promedio Real
EDT	Estructura de Descomposición del Trabajo
EPE	Esfuerzo Promedio Estimado
EPR	Esfuerzo Promedio Real
FiSMA	Finnish Software Measurement Association
FPA	Function Point Analysis
FSM	Functional Sizing Methods
GQM	Goal/Question/Metric
IFPUG	International Function Point Users Group
ISBSG	International Software Benchmarking Standards Group
LACCIR	Latin American and Caribbean Collaborative ICT Research
LOC	Lines Of Code
MPS.BR	Mejora del Proceso de Software Brasileño

NESMA	Netherlands Software Metrics Association
PESC-ESE	Programa de Engenharia de Sistemas y Computacion - Experimental Software Engineering
PMBOK	Project Management Body of Knowledge
PPED	Porcentaje de Precisión de la Estimación de la Duración
PPEE	Porcentaje de Precisión de la Estimación del Esfuerzo
RCC-TIC	Red Colombiana de Calidad para el Desarrollo en Tecnologías de Información y Comunicación
SWEBOK	Software Engineering Body of Knowledge
UCP	Use Case Point
UKSMA	United Kingdom Software Metrics Association
WBS	Work Breakdown Structure

Capítulo 1

Introducción

1.1. Motivación

Realizar una adecuada estimación de la duración y el esfuerzo que va a ser invertido en el desarrollo de un proyecto de software, es uno de los retos más importantes que enfrentan las organizaciones de software que están iniciando un camino de mejora. A pesar de que en la literatura se han publicado un conjunto de métodos para realizar la estimación de la duración y el esfuerzo de los proyectos de software, se percibe que las Pymes realizan esta estimación de manera intuitiva.

Las pequeñas y medias empresas, Pymes, de software que inician proyectos de mejora de sus procesos con el objetivo de lograr mayores niveles de madurez, se enfrentan con el desafío de seleccionar, de una parte, el método de estimación más adecuado a su contexto y, de otra parte, adoptar mejores prácticas que permitan ir consolidando una base histórica de estimación con el propósito de disminuir cada vez más la desviación entre los valores estimados y reales. [39].

Existen en la literatura técnica diversos métodos de medición del tamaño del software que sirven para calcular el tamaño de una aplicación de software y a partir de estos datos estimar el esfuerzo de desarrollo y pueden ser clasificados en tres categorías: métodos de medición del tamaño funcional aceptados oficialmente por la ISO/IEC, [34], [48], métodos paramétricos, son aquellos en los cuales el proceso de cuantificación del resultado es basado en un proceso mecánico, por ejemplo la aplicación de una formula derivada de los datos históricos [7], [11], [40]. Por último los métodos heurísticos que son aquellos en los cuales la cuantificación del resultado es producida a partir del juicio y/o la experiencia de un experto [45], [69]. O una combinación de varios métodos para alcanzar una mejor precisión en las estimativas.

Ciudad	Número de Empresas	Porcentaje de Participación
Bogotá	23	45 %
Medellín	13	25 %
Cali	9	18 %
Bucaramanga	6	12 %
Total	51	100 %

Tabla 1.1: Distribución de las empresas participantes en el proyecto.

Los diferentes modelos de calidad, recomiendan la aplicación de prácticas que favorecen la estimación adecuada de los proyectos de desarrollo o mantenimiento de software. En el caso particular de CMMI-DEV v1.2, estas prácticas se agrupan en dos categorías: (a) prácticas específicas asociadas a la planificación del proyecto (área de procesos PP - Project Planning) y al monitoreo y control del proyecto (área de procesos PMC - Project Monitoring and Control); (b) prácticas genéricas que crean las condiciones organizacionales para una adecuada apropiación (institucionalización) del proceso por parte de los equipos de trabajo.

Existen diversos investigadores han estudiado el problema de la estimación de software en las compañías de software en diferentes países [10], [11], [17], [43], [45], [55] pero de acuerdo la recopilación bibliográfica realizada para este estudio ninguna investigación de este tipo ha sido efectuada en Colombia.

Desde el año 2005, el gobierno Colombiano ha venido apoyando la mejora de procesos de las empresas de software; utilizando CMMI como modelo referente; durante el período 2008-2010 se realizó un proyecto a nivel nacional que acompañó 51 empresas en todo el país para la adopción del modelo CMMI-DEV v1.2 y 18 de estas empresas recibieron apoyo económico para realizar el proceso de valoración oficial SCAMPI, [2].

La población de este estudio acerca de la estimación fue tomado del conjunto de Pymes que formaron parte de este proyecto y por lo tanto se espera que ya tengan cierto nivel de madurez en la manera de realizar la estimación de sus proyectos de desarrollo de software. La Tabla 1.1, muestra la distribución de empresas vinculadas al estudio experimental y su distribución en las principales ciudades de Colombia.

La realización de este tipo de estudios experimentales en empresas colombianas, abre un camino interesante que va permitir recopilar evidencias del estado actual de la industria de software y del impacto que puede tener los programas de mejora de procesos adelantados de manera individual por cada organización o realizados de manera masiva dentro de un esquema de financiación.

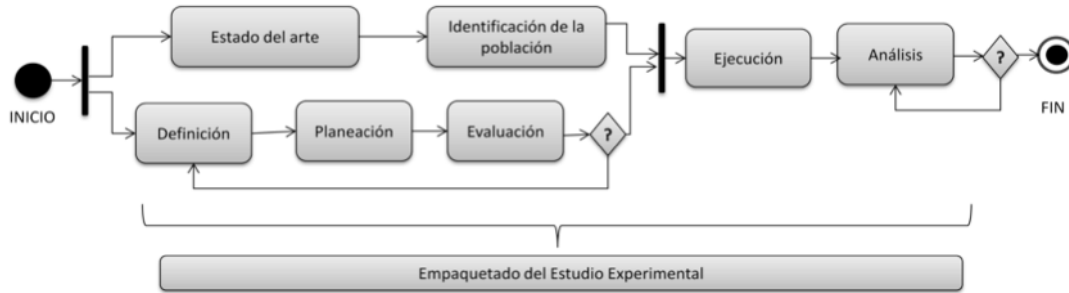


Figura 1.1: Metodología de Trabajo, adaptado de [64].

1.2. Objetivos

El objetivo de este estudio es analizar la brecha que existe entre la teoría y la práctica con respecto a la estimación de proyectos de software, considerando dos perspectivas:

- Un primer análisis pretende caracterizar las prácticas de cada empresa y el grado de aplicación respecto a las prácticas recomendadas por el modelo CMMI-DEV v1.2.
- Un segundo análisis busca determinar cuáles son los métodos de estimación más usados en la industria y su porcentaje de precisión en las estimaciones de duración y esfuerzo, según datos históricos proporcionados por las empresas.

1.3. Metodología de Trabajo

Para alcanzar los objetivos descritos en esta investigación, se aplicó una metodología de Estudio Experimental propuesta por Guilherme H. Travassos en [64], y en base a esta, fue adaptada y se definió el siguiente flujo de trabajo, presentado en la Figura 1.1.

El estudio del estado del arte de la estimación de software consistió en una búsqueda en la literatura científica para identificar cuáles han sido los temas más comúnmente investigados respecto a la estimación de software. Esta búsqueda literaria se enfocó a los estudios relacionados a la estimación del esfuerzo y la calendarización de un proyecto de software.

En paralelo se comenzó a trabajar en la etapa de definición del estudio experimental siguiendo el protocolo definido por [64], el cual plantea que durante esta etapa se especifican los objetivos del estudio, preferiblemente bajo un enfoque Goal/Question/Metric (GQM) descrito por [53], tanto globales como específicos. También se define el foco de calidad del

estudio, la perspectiva desde la cual será llevado a cabo, la selección del contexto y se determinan las preguntas de investigación.

Una vez definido el estudio, se procede a la etapa de planeación, en la cual se llevaron a cabo actividades para definir las hipótesis del estudio, la selección de las variables, tanto dependientes como independientes, se determina el personal, los recursos, el diseño del experimento, la instrumentación a utilizar, los mecanismos de análisis y los criterios para evaluar la validez de los resultados. A continuación se realizó una evaluación de la propuesta del estudio, para determinar la pertinencia y conveniencia de realizarlo, en caso que no sea aprobado, se procederá a rehacer la etapa de definición y planeación nuevamente. En caso positivo, que el estudio sea aprobado, se puede proceder con la fase de ejecución del estudio experimental.

En paralelo, se trabajó trabajamos en la identificación de la población objetivo; para tal propósito se contó con la colaboración de la Red Colombiana de Calidad para el Desarrollo en Tecnologías de Información y Comunicación RCC-TIC [54] y se solicitó el diligenciamiento de una encuesta al grupo de empresas que participaron en un ciclo de mejora de la calidad utilizando CMMI-DEV v1.2, en el marco de un proyecto liderado por RCC-TIC [54].

La ejecución del estudio experimental se llevo a cabo en los meses de Noviembre y Diciembre de 2011. Por medio de una encuesta que fue diligenciada por un representante de la empresa. Esta encuesta contiene los aspectos más relevantes a ser analizados respecto a las prácticas de estimación de software y el porcentaje de error en la estimación del esfuerzo y de la calendarización en los proyectos de software. Ver Anexo A.

A continuación se procedió a realizar el análisis, se llevo a cabo una depuración de los datos encontrados, intentando identificar tendencias y evidencias para apoyar o refutar las hipótesis, y de esta manera alcanzar los objetivos formulados inicialmente.

Por último se realizo el proceso de empaquetado del estudio experimental, en el cual se consolidaron las conclusiones encontradas en el estudio, se identifico el nivel de confianza alcanzado y como resultado final se realizó un artículo el cual fue aceptado en el congreso ESELAW '12 Experimental Software Engineering Latin American Workshop 2012, en el cual se presentaron los principales resultados del estudio [49].

1.4. Organización

A demás del capítulo de Introducción este trabajo contiene otros cinco capítulos organizados de la siguiente manera:

Capítulo 2. Conceptos y Trabajos Relacionados: presenta las diferentes categorías de los métodos de estimación de software y en qué consiste cada uno. También presenta una visión del estado del arte de los estudios sobre estimación de software y que se han sido publicados, los cuales sirven para tener un referente internacional acerca de los estudios empíricos realizados en el tema de estimación.

Capítulo 3. Estudio Experimental en Estimación de Software: presenta el protocolo del estudio experimental, para las etapas de definición y planeación.

Capítulo 4. Análisis de los Datos Recolectados: presenta en detalle el análisis que se llevo a cabo sobre los datos encontrados de acuerdo al protocolo definido, con el propósito de responder las preguntas de investigación, validar hipótesis e identificar tendencias.

Capítulo 5. Conclusiones: presenta las contribuciones del trabajo, limitaciones, publicaciones asociadas a este trabajo de maestría y discute las perspectivas de los trabajos futuros.

1.5. Síntesis de los resultados alcanzados

Dentro de los resultados alcanzados de este trabajo, se destacan los siguientes:

- La elaboración y ejecución de un estudio experimental primario con énfasis en la estimación de software en las Pymes colombianas que realizaron un primer ciclo de mejora con CMMI-DEV v1.2.
- La vinculación de 13 Pymes del área de software, representando las cuatro mayores ciudades de Colombia, Bogotá, Medellín, Cali y Bucaramanga, las cuales contribuyeron a la realización de este estudio proveyendo información real de sus prácticas de estimación así como de la precisión en la estimación del esfuerzo y el calendario de los proyectos de software.
- La realización de una estancia investigativa por un periodo de 3 meses, en el Grupo de Investigación en Ingeniería de Software Experimental PESC-ESE de COPPE/UFRJ. Allí se estudiaron temas en el área de Ingeniería de Software Experimental y el reconocimiento de los diferentes tipos de estudios: estudios primarios, estudios secundarios (tales como revisiones sistemáticas), y estudios terciarios (tales como agregación). Al igual se estudió acerca de la mejora de procesos de software tomando como referente el Modelo Brasileiro de Mejora de Calidad de Software, denominado MPS.Br [66].
- Elaboración de un artículo llamado Estado actual de la estimación de software en compañías colombianas que han adoptado CMMI, en coautoría con Raquel Anaya y Guilherme H. Travassos, asesores del proyecto de grado. Este artículo fue presentado en el Congreso Latinoamericano de Software Experimental ESELAW 2012. [49]

Capítulo 2

Conceptos y Trabajos Relacionados

2.1. La naturaleza de la estimación de proyectos de software.

De acuerdo al diccionario americano [1], una estimación se define como “1. Una evaluación tentativa o un cálculo preliminar. 2. Un cálculo preliminar del costo de un proyecto. 3. Un juicio basado en unas impresiones; opiniones”. Por su parte, el Diccionario de la Real Academia Española (RAE) da una definición de estimación en un contexto general “Aprecio y valor que se da y en que se tasa y considera algo” y otra definición particular en el marco del derecho “estimación que se realiza en ciertos tributos para determinar el valor de la base imponible.

Siguiendo la idea planteada por McConnell [41], cuando en el contexto de los proyectos de software se habla de estimación, es poco probable que se entienda como un cálculo tentativo preliminar que se espera que cambie más tarde: generalmente para los ejecutivos y clientes del proyecto la estimación está asociada con objetivos de negocio, acuerdos o planes, llegando incluso a un compromiso que, en caso de no cumplirlo, puede tener consecuencias legales.

En esta sección se analizan algunas características de la estimación de proyecto de software que si bien son el principio orientador de las mejores prácticas articuladas en referentes de calidad como CMMI, son difíciles de aplicar en los contextos cambiantes de proyectos de software. [41].

2.1.1. Relación entre la estimación y el plan del proyecto.

Estos dos conceptos están directamente relacionados, pero es importante entender su diferencia. Mientras que el objetivo de la medición es la precisión, el objetivo del plan es alcanzar unas metas. Si bien los estimados dan el soporte para el plan, éste debe tener en cuenta que si el nivel de confianza de la estimación es baja, el plan necesita reconocer el hecho y asumir un adecuado nivel de riesgo en el cumplimiento de las metas. Este es el principio que permite entender los factores de corrección que se encuentran involucrados en los diferentes modelos de estimación, buscando definir un margen de seguridad entre la desviación de estimación y las metas establecidas en el plan. [41].

2.1.2. Un buen nivel de confianza en la estimación representa una ventaja competitiva para la organización.

Una estimación ajustada a la realidad es deseable en cualquier aplicación de software, no solamente para definir el presupuesto, los recursos, tiempos, costes y para evitar el sobre esfuerzo en los equipos de trabajo, sino también para alcanzar una ventaja competitiva, ya que las organizaciones de software con estimaciones concretas y ajustadas a la planificación serán capaces de obtener los proyectos en licitación. [45].

Una adecuada estimación inicial de un proyecto de software es de vital importancia a la hora de cerrar un negocio para una compañía, ya que es por medio de esta que se rige el buen funcionamiento y el éxito de un proyecto. Una mala estimación inicial puede acarrear consecuencias negativas para una organización.

Si una empresa sobre estima el tamaño y la complejidad de un proyecto, puede encontrarse en desventaja competitiva y existe la posibilidad que otra empresa presente una estimación más ajustada a la realidad con plazos más cortos y costos más bajos, la cual al final será obtendrá el negocio.

En el caso contrario, cuando una empresa subestima el tamaño y la complejidad de un proyecto intentado ser atractiva ante una eventual licitación, existe el riesgo que este proyecto rebase la capacidad productiva de la empresa, lo cual afecte su nivel de cumplimiento con los entregables del proyecto y al término final del proyecto llevarla a una crisis financiera.

La estimación como ventaja competitiva para una organización, puede a su vez ser perjudicial cuando se trata de estudios como el que se intenta realizar en este trabajo. Las empresas que han llegado a estabilizar un método de estimación, no están siempre dispuestas a compartir en este tipo de estudios, “los secretos” que las hacen competitivas.

2.1.3. La estimación es tanto arte como ciencia

Buena parte de los esfuerzos de investigación en este tema, se orientan a proponer nuevos métodos de estimación o a plantear mejoras en los métodos existentes que garanticen un mayor nivel de confiabilidad. Cuando las organizaciones de software ya ha adquirido cierto nivel de madurez, pueden tener como meta mejorar sus estimados para pasar de una desviación del $\pm 10\%$ al $\pm 5\%$; de otra parte, en la mayoría de organizaciones de software su principal preocupación es evitar desviaciones de estimados mayores al 100% . [41].

Teniendo en cuenta los múltiples factores de contexto involucrados en un proyecto software, un método de estimación, por más complejo que sea, no puede garantizar el mejor resultado en todos los contextos. Es por eso que un proceso adecuado de estimación en una organización, debe buscar un equilibrio entre la formalidad y rigurosidad de un método de estimación y prácticas de sentido común y flexibilidad que permiten identificar factores de contexto no controlados.

Se podría quizás afirmar que esta doble naturaleza de la estimación como arte y ciencia, puede ayudar a explicar las tres principales tendencias de los métodos de estimación que se presentarán más adelante: métodos heurísticos (principalmente arte puesto que están soportados en el sentido común y la experiencia), métodos paramétricos (principalmente ciencia puesto que están fundamentados en modelos matemáticos) y métodos de estimación de tamaño funcional (como una combinación que busca un equilibrio entre ciencia y arte). [41].

2.1.4. Influencia de la dinámica del proyecto en las estimaciones

Como ya se mencionó, son tan diversos los factores involucrados en un proyecto de desarrollo de software, ver Figura 2.1, que es muy poco probable lograr una estimación con una precisión del 100% . La precisión de la estimación, estará en la mayoría de ocasiones influenciada por la fase dentro del proyecto en que ésta se realice. McConnell llama a este fenómeno “la incertidumbre del cono” que afirma que la precisión de los estimados será mayor a medida que avanza el proyecto [41].

Esta dinámica planteada en la Figura 2.1, ya que a lo largo del proyecto pueden aparecer nuevos requisitos, eliminar requisitos, cambio en el personal del proyecto, etc, . Lo cual permite entender que la forma misma de aplicación de los métodos de estimación en una organización puede variar, dependiendo de qué tan flexible sea el proyecto para permitir procesos de re-estimación intermedios.

Teniendo en cuenta que el foco de este estudio está orientado a organizaciones de software que desarrollan soluciones software para tercero, se observan algunos de los escenarios en

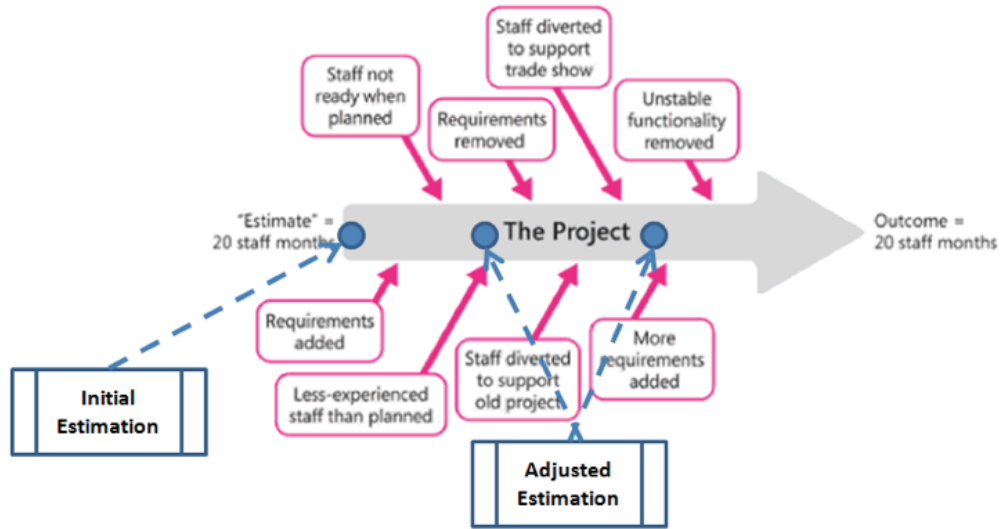


Figura 2.1: Influencia de la dinámica del proyecto en las estimaciones. Adaptado de [41].

favorecen esta flexibilidad en la estimación:

Escenario 1. Algunas organizaciones han logrado realizar contrataciones con los clientes en modalidad por fases y no a tiempo y costo fijo (por ejemplo se contrata primero la fase de Elaboración y luego se hace una reestimación para la construcción), lo cual disminuye el riesgo de una estimación temprana.

Escenario 2. En algunas organizaciones se establece una relación (cliente-proveedor) tan estrecha y de tanta confianza, que el proveedor es un aliado estratégico para que el cliente alcance sus objetivo de negocio. En este caso las estimaciones y planes se realizan de manera conjunta y los equipos de trabajo establecen dinámicas de trabajo de estilo ágil, en donde la estimación está orientada a tareas de granularidad más fina a través de iteraciones; generalmente es estos esquemas se utilizan enfoques de gestión tipo SCRUM con estimaciones basadas en Sprint, como se explicará más adelante.

Escenario 3. En algunos escenarios (por ejemplo cuando se contrata con entidades del estado), se debe realizar un propuesta del proyecto, definiendo tiempo y costo fijo a partir de unos términos de referencia (RFP - Requesto For Proposal), que muchas veces no tienen el nivel de detalle requerido para una estimación confiable. En este escenario, contar con un modelo confiable que permita hacer una buena estimación inicial antes de ofertar un proyecto, puede ayudar a la organización a detectar si el proyecto el viable de abordar.

En situaciones en que la parte comercial hace compromisos de tiempos y costos de proyecto, contar con un modelo de estimación previa temprano permitiría detectar el riesgo de la planificación y serviría para enfrentar dicho riesgo de manera temprana y no al final del plazo, cuando ya hay muy poco margen de maniobra.

2.1.5. La importancia de los datos históricos como base de estimación.

La carencia o difícil acceso a datos históricos de proyectos de software que puedan servir de referencia a los gerentes de proyecto y personal administrativo, es uno de los principales obstáculos de las empresas a la hora de realizar los estimativos para un nuevo proyecto. Es de ahí que nace la importancia de los datos históricos como base de estimación, para brindar luz a los responsables de esta práctica dentro de la organización y mostrar evidencias tangibles de estimaciones de proyectos anteriores.

Como ejemplos internacionales de estas bases de conocimiento de encontramos el ISBSG, International Software Benchmarking Standards Group, la cual es una organización global e independiente encargada de proveer datos y análisis para la industria de TI. ISBSG actualmente cuenta con dos repositorios de datos.

El primer repositorio de datos se trata de proyectos de desarrollo de software y mejoras, el cual contiene más de 5.600 proyectos, los cuales han sido recolectados desde el año 2001. Un segundo repositorio es de proyectos de soporte y mantenimiento el cual cuenta con más de 490 aplicaciones, cuyos datos han sido recopilados desde el año 2008, siendo este repositorio mas reciente que el anterior. ISBSG es una organización mundial la cual tienen empresas aliadas en 22 países, las cuales brindan información detallada y estandarizada de los proyectos llevados a cabo, [47].

Como ejemplo del uso de esta base de conocimiento Capretz et al, en [11] Propone un nuevo método de estimación llamado Voting framework. Basado en los resultados obtenidos de la comparación de 5 métodos de estimación paramétricos, tomando como insumos 439 proyectos descritos en ISBSG que se adaptaban a las características específicas de su estudio.

Otro ejemplo de organizaciones encargadas de recolectar la información que permita mejorar el desempeño de las compañías de IT, administrando mejor los recursos de software, lo encontramos en el CSBSG , Chinese Software Benchmarking Standards Group. Esta base de conocimiento de las empresas de TI chinas, fue analizada por Da Yang et al, en [19] donde intento identificar posibles áreas de mejora en la estimación de software, analizando 112 proyectos y realizando 116 encuestas a empresas. Como resultados de su análisis encontró que proyectos grandes son más propensos a sobre esfuerzo y demora en la entrega.

Es bueno resaltar que la credibilidad del resultado de un estudio, reside en gran parte en la credibilidad de los datos de entrada del mismo y la imparcialidad del análisis. Un caso negativo, el cual se convierte en un ejemplo a no seguir lo encontramos en el famoso CHAOS Report publicado por el Standish Group, El cual trata acerca de la estimación del esfuerzo y la duración de los proyectos en las compañías americanas de software. Sin embargo se descarta su inclusión en este estudio debido a problemas de validez como se lo evidenció Jorgensen et al. en [36].

2.2. Métodos de estimación de software

En la actualidad existe gran variedad de métodos utilizados para estimar y medir el tamaño de una aplicación de software. A continuación se presenta una síntesis de estos métodos agrupados en tres categorías: métodos de medición del tamaño funcional, métodos de estimación basados en parámetros y métodos de estimación heurísticos o basados en la experiencia.

2.2.1. Métodos de medición de tamaño funcional

Es la categoría de métodos en los cuales más se ha trabajado con el propósito de definir estándares para la industria. Están soportados en el principio de definir la estimación dependiendo de las características funcionales del producto software. Actualmente hay cinco estándares para la medición del tamaño funcional del software (FSM) reconocidos en la norma ISO/IEC 14143-1:2007 Information technology. Software measurement. Functional size measurement. Part 1: Definition of concepts. [34], [48]:

ISO/IEC 20926: IFPUG 4.1 (2003)

Método definido en la norma ISO/IEC 20926:2009 Software and systems engineering - Software measurement - IFPUG functional size measurement method, [32]. Consiste en asignar una cantidad de puntos a una aplicación de software según la complejidad de los datos que maneja y de los procesos que realiza sobre ellos, considerándolo siempre desde el punto de vista del usuario. Este método sirve para calcular el tamaño de una aplicación de software y requiere desarrollar una serie de actividades tales como: Determinar el tipo de aplicación a medir, identificar el alcance de lo que se va a medir y los límites de la aplicación, contar las funciones de datos, contar las funciones transaccionales, calcular el recuento bruto de puntos función.[31] , [32].

Cabe mencionar que este método IFPUG 4.1 como estándar no incluye a la parte de ajuste de puntos de función, ya que este ajuste no reúne los requisitos necesarios de la ISO/IEC

14143-1. El método IFPUG 4.1 asume un modelo de software que consta de dos partes: Primero, el Modelo de Datos que representa a los archivos lógicos internos y externos. Segundo, El Modelo de Transacciones que representa las operaciones que el usuario desea hacer con los datos. Estos pueden ser de entrada, salida y consulta. [15].

ISO/IEC 20968: Mk II FPA (2002)

Método definido en la norma ISO/IEC 20968:2002 Mk II Function Point Analysis - Counting Practices Manual, [42], y adoptado por la United Kingdom Software Metrics Association (UKSMA). Intenta ser un método de medición continua a lo largo del ciclo de vida de una aplicación, frente a unas mediciones más estáticas que ofrecen FP y FPA. Consiste en identificar los requisitos funcionales del software; cada uno de ellos se clasifica en uno de los siguientes tipos: entradas, salidas y objetos. Después se realiza un conteo con el fin de determinar el tamaño del sistema en función de estos requisitos funcionales.

Para lograr esto, es indispensable determinar los límites del sistema, y trazar la línea que separa la lógica, de los usuarios del sistema.

Esto con el fin de determinar las operaciones lógicas tales como entradas y salidas, las cuales son aquellas que cruzan los límites de la aplicación, durante la interacción entre el usuario y el sistema. Una vez que las operaciones y los objetos del sistema han sido identificados, estos pueden ser contados con el fin de encontrar el tamaño funcional del sistema. El tamaño funcional del sistema se representa como recuentos ponderados de entrada y/o salida transacciones y objetos dentro de los límites del sistema. Finalmente el tamaño de aplicación es la suma de los tamaños de las operaciones lógicas, cada transacción se cuenta una vez, aunque pueda ser ejecutada en más de un punto de la aplicación. [42], [68].

ISO/IEC 19761: COSMIC -FFP (2003)

Método definido en la norma ISO/IEC 19761:2003 COSMIC Common Software Measurement International Consortium- A Functional Size Measurement Method. [17]. Este método está basado en el principio de que los requisitos funcionales de los usuarios de software son generalmente procesos funcionales. Cada proceso funcional se activa cuando un usuario del software (una persona, un dispositivo de hardware o software) reconoce un evento y envía un mensaje para iniciar el proceso. El proceso se completa cuando el software ha hecho todo lo que se requiere para responder al evento.

Algunos ejemplos de eventos de activación de procesos funcionales de software podrían ser:

En una aplicación de negocio:

- Se recibe un pedido - Introducir una orden
- Un empleado se casa - Actualizar registro de personal
- Fin de mes - Producir estados de cuenta bancarios

En una aplicación en tiempo real:

- Avance en el reloj - Inicio del ciclo para leer instrumentos y ajustar el control de un proceso
- Comando de un piloto de avión - subir plataforma de aterrizaje en el despegue
- Recepción de un mensaje telefónico - Realizar una llamada telefónica

Más adelante se toman estos procesos funcionales y se analizan como movimientos de datos. El conteo de los movimientos de datos (dentro y fuera del software, y hacia y desde el almacenamiento persistente) se toma como la medida del tamaño funcional del software, en unidades de Puntos de Función Cósmica, FPC. [16], [17].

En resumen, COSMIC-FFP distingue cuatro tipos de movimientos de datos: entrada, salida, lectura y escritura. Un movimiento de dato transfiere uno o más atributos de datos que pertenecen a un único grupo de dato. Así mismo, COSMIC-FFP establece como principio de medición que el tamaño funcional del software es directamente proporcional al número de movimientos de datos.[15].

ISO/IEC 29881: FiSMA FSM (2008)

Método definido en la norma, ISO/IEC 29881:2008 Information technology - Software and systems engineering - FiSMA 1.1 functional size measurement method. [23]. El proceso de medición del método FISMA FSM se compone de dos partes principales: la medición de los servicios de interfaz de usuario final y la medición de los servicios indirectos. Si una de estas dos partes no existe en la pieza de software, entonces el proceso consiste sólo en la medición de los servicios que están presentes. Esta medición se realiza por medio de un conteo de los Componentes Funcionales Base (BFC) la cual es una unidad primaria de los requisitos funcionales, definida por un método de FSM para fines de medición.

De esta forma los BFC se agrupan en 7 categorías diferentes: [22]

- Interactividad del usuario final y servicios de consulta (q)
- Interactividad de entradas del usuario final (i)
- Sin interactividad con el usuario final y servicios de salida (o)
- Interfaces de servicios hacia otras aplicaciones (t)

- Interfaces de servicios desde otras aplicaciones (f)
- Servicios de almacenamiento de datos (d)
- Algoritmos y manipulación de servicios (a)

Estas categorías se descomponen a su vez en Tipos de BFC, donde cada pieza de software puede incorporar varios y/o diferentes requisitos de los usuarios funcionales, en total existen 28 Tipos de BFC. Las normas de conteo de cada tipo de BFC son definidas en el capítulo 5 de [24]. El proceso detallado para la cuenta del tamaño total de la aplicación de software se explica en el capítulo 7 de [24].

ISO/IEC 24570: NESMA FPA (2005)

Método definido en la norma ISO/IEC 24570:2005 Software engineering - NESMA function size measurement method version 2.1 - Definitions and counting guidelines for the application of Function Point Analysis [47], Es un método que al igual que el IFPUG está basado en el Análisis de Puntos de Función FPA, el cual se utiliza para determinar el tamaño funcional de un sistema de información o proyecto. El tamaño funcional puede ser utilizado para diferentes propósitos, por ejemplo estimación de calendario, presupuesto, etc.

NESMA FPA propone 5 pasos para calcular el tamaño funcional del sistema:

Paso 1. Identificar las funciones del sistema que son relevantes al usuario. Para esto FPA tiene 5 categorías de funciones de usuarios:

- Archivo Interno Lógico (Internal Logical File, ILF)
- Archivo de Interface Externa (External Interface File, EIF)
- Entrada Externa (External Input, EI)
- Salida Externa (External Output, EO)
- Consulta Externa (External Inquiry, EQ)

Paso 2. Determinar la complejidad de cada función. El nivel de complejidad que puede tomar los siguientes valores:

- Bajo (Low)
- Promedio (Average)
- Alto (High)

Paso 3. Calcular el Total de Puntos de Función sin ajustar, se realiza una vez los pasos 1 y 2 han sido ejecutados, y se le asigna un valor a cada funcionalidad de acuerdo a la Tabla 2.1.

	Nivel de Complejidad		
Tipo de Función	Bajo	Promedio	Alto
Archivo Interno Lógico	7	10	15
Archivo de Interface Externa	5	7	10
Entrada Externa	3	4	6
Salida Externa	4	5	7
Consulta Externa	3	4	6

Tabla 2.1: Matriz de valores de FP sin ajustar. Tomado de [46].

Características generales del sistema para FPA	
1. Comunicación de datos	8. Actualización en línea
2. Procesamiento de datos distribuidos	9. Procesamiento complejo
3. Desempeño	10. Reusabilidad
4. Configuración Altamente Usada	11. Facilidad de instalación
5. Tasa de la Transacción	12. Facilidad de uso
6. Entrada de datos en línea	13. Portabilidad
7. Eficiencia usuario final	14. Facilidad de cambio

Tabla 2.2: Características generales del sistema para FPA. Tomado de [46].

Paso 4. Calcular el peso de los requisitos funcionales usando las 14 características del sistema para el análisis FPA. Ver Tabla 2.2.

Cada una de estas características es evaluada en un rango desde 0 (No influencia) a 5 (Mucha influencia), basado en un criterio pre-establecido. Este criterio puede ser encontrado en la norma [47], El resultado total, también llamado Grado Total de Influencia tiene un valor entre 0 (0 x 14) y 70 (5 x 14).

Paso 5: Calcular la cuenta de Puntos de Función Ajustados del sistema. Utilizando el Grado total de influencia encontrado en el paso anterior, se calcula la siguiente fórmula:

$$\text{Valor del Factor de Ajuste} = 0.65 + 0.01 * \text{Grado Total de Influencia}$$

Por consiguiente para calcular el valor de puntos de función ajustados, se utiliza la siguiente fórmula:

$$\text{Total Puntos de Función Ajustados} = \text{Puntos de Función Sin Ajustar} \times \text{Valor de Factor de Ajuste}$$

En conclusión, el Valor del Factor de Ajuste puede incrementar o disminuir el número de puntos de función, a lo sumo en 35 % (0.65 + 0.00) o (0.65 + 0.70). En la práctica, esta corrección varía entre el - 10 % y + 10 %. Si el Valor del Factor de Ajuste es 1.00,

métodos muy útiles cuando se desarrollan proyectos similares unos a otros, proyectos con la misma tecnología, o proyectos en áreas funcionales similares. Entre los métodos más conocidos en esta categoría se pueden mencionar:

Puntos de Casos de Uso (UCP)

Está basado en los tradicionales métodos de Puntos de Función descritos previamente. La principal ventaja de este método es su adaptación en empresas o proyectos que utilizan la técnica de los casos de uso. Generalmente consiste de cinco pasos fundamentales, [12], [3]:

Paso 1. Cálculo de los Puntos Caso de Uso sin ajustar (UUCP).

Paso 2. Cálculo de los factores técnicos (TCF).

Paso 3. Cálculo de los factores de entorno (EF)

Paso 4. Cálculo de los Puntos Caso de Uso ajustados (UCP) y

Paso 5. Estimación del esfuerzo.

Como ocurre en otros métodos de estimación, una vez obtenido el tamaño, se puede obtener el esfuerzo, que estará dada en unidades horas-hombre [12], mediante la siguiente expresión:

$$Esfuerzo = UCP * Factor de Productividad$$

COCOMO, COCOMO II

(Constructive Cost Model), es un método de estimación de software basado en algoritmos desarrollado por Barry W. Boehm, et al. [7] . Este método tiene como objetivo estimar el esfuerzo, costo y calendario de un proyecto de software, utiliza una fórmula de regresión básica con los parámetros que se derivan de datos de proyectos históricos y características actuales del proyecto. Su primera versión COCOMO (1981) tomo como unidades de medida las líneas de código (LOC), porque en su momento predominaban los proyectos que utilizaban lenguajes estructurados tales como: assembly o PL/I. Más adelante en 1995, sale una nueva versión COCOMO II, el cual se adapta mejor a la estructura de los proyectos actuales, basados en lenguajes de programación orientados a objetos [14].

El modelo COCOMO II está conformado por dos partes: una aplicable al comienzo de los proyectos llamado Diseño preliminar (Early Design) y la segunda parte aplicable luego del establecimiento de la arquitectura del sistema, post-arquitectura, (Post Architecture). El modelo posee un conjunto de 17 estimadores de costo, ver Tabla 2.3, y un conjunto de 5

	Estimadores de costo	Valor	Descripción
1	DATA	alto	Tamaño de la base de datos
2	CPLX	nominal	Complejidad del producto
3	TIME	nominal	Limitación en tiempo de ejecución
4	STOR	nominal	Limitaciones en almacenamiento principal
5	RUSE	nominal	Usabilidad requerida
6	DOCU	nominal	Documentación consistente con necesidades del ciclo de vida.
7	PVOL	nominal	Portabilidad
8	SCED	Muy alto	Factor de calendarización
9	RELY	nominal	Confiabilidad requerida
10	TOOL	nominal	Uso de herramientas de software
11	APEX	nominal	Aplicación de la experiencia
12	ACAP	bajo	Capacidad del analista
13	PCAP	alto	Capacidad del programador.
14	PLEX	bajo	Experiencia en la plataforma
15	LTEX	alto	Experiencia en el language de programacion y herramientas
16	PCON	nominal	Continuidad del personal
17	SITE	nominal	Desarrollo Multiplataforma

Tabla 2.3: Estimadores de costo (Cost Drivers) COCOMO II. Tomado de [60]

factores de escala que afectan de manera exponencial en el esfuerzo del proyecto, ver Tabla 2.4. [60].

Para el calculo final del tamaño del sistema, COCOMO II, brinda herramientas que permiten hacer estos cálculos automáticos, la Figura 2.3, muestra un ejemplo de estas [14].

Técnicas basadas en Regresión

Estas técnicas están orientadas a encontrar funciones matemáticas a partir de los datos de proyectos históricos. Son un conjunto de técnicas utilizadas generalmente para reemplazar los valores perdidos en los datos. La variable de los datos faltantes se trata como la variable dependiente, mientras que el resto de los casos se tratan como variables independientes. Una ecuación de regresión se genera cada vez que se puede utilizar para predecir los valores

	Factores de escala	Valor	Descripción
1	PREC	nominal	Precedencia
2	PMAT	CMM Nivel 2 (o más)	Madurez del proceso
3	TEAM	nominal	Sinergia del equipo
4	FLEX	nominal	Flexibilidad de desarrollo
5	RESL	poco (20 %)	Arquitectura y solución de riesgos

Tabla 2.4: Factores de escala (scale factors) COCOMO II. Tomado de [60]

The screenshot shows the COCOMO II - Constructive Cost Model web application. The interface includes a header with the NPS logo and the title "COCOMO II - Constructive Cost Model". Below the header, there are several sections for inputting project data:

- Software Size:** Includes a "Sizing Method" dropdown set to "Source Lines of Code" and a "SLOC" section with input fields for "New", "Reused", and "Modified" code.
- Software Scale Drivers:** Includes dropdowns for "Precedentedness", "Development Flexibility", "Architecture / Risk Resolution", "Team Cohesion", "Process Maturity", and "Unfamiliarity".
- Software Cost Drivers:** Includes dropdowns for "Required Software Reliability", "Data Base Size", "Product Complexity", "Developed for Reusability", "Documentation Match to Lifecycle Needs", "Personnel" (Analyst Capability, Programmer Capability, Personnel Continuity, Application Experience, Platform Experience, Language and Toolset Experience), "Platform" (Time Constraint, Storage Constraint, Platform Volatility), and "Project" (Use of Software Tools, Multisite Development, Required Development Schedule).
- Software Labor Rates:** Includes a "Cost per Person-Month (Dollars)" input field and a "Calculate" button.

On the right side, there is a "Model(s)" dropdown set to "COCOMO", and checkboxes for "Monte Carlo Risk" and "Auto Calculate", both set to "Off".

Figura 2.3: COCOMO II - Constructive Cost Model. Tomado de [14].

que faltan. Estos métodos reducen la variación asociada con otras técnicas.

Como ejemplos de estos métodos encontramos [11]:

- Regresión estandar (Standard regression - Ordinary Least Squares (OLS) method)
- Regresión robusta (Robust Regression)

Métodos basados en técnicas de inteligencia artificial

Estos métodos se basan en el uso de datos recogidos en proyectos anteriores en los que se realizaron estimaciones y utilizan diferentes técnicas de extracción de conocimiento basadas en principios de la inteligencia artificial; buscan realizar estimaciones mas eficientes, eficaces y, si fuera posible, con mayor precisión, que las realizadas por los métodos paramétricos. Dentro de las técnicas que existen en esta categoría se pueden mencionar las redes neuronales, las técnicas orientadas al aprendizaje, el razonamiento basado en casos, los árboles de regresión, los agentes dinámicos, programación genética y lógica difusa. [18]. De estos métodos, el que aparece mencionado en los trabajos empíricos estudiados es el de lógica difusa.

El método de lógica difusa (Fuzzy Logic), es un método de estimación utilizado para calcular el tamaño del proyecto en líneas de código. Consiste en estimadores que clasifican las funcionalidades en cinco categorías: Muy Pequeña, Pequeña, Mediana, Grande y Muy Grande. A continuación, pueden utilizar los datos históricos de proyectos sobre el número de líneas de código que las funcionalidades de categoría Muy Pequeña requieren, el número de líneas de código que las funcionalidades de categoría Pequeña requieren, y así sucesivamente para calcular los totales de las líneas de código [11], [25], [40].

2.2.3. Métodos de estimación basados en la experiencia o heurísticos

Es el grupo de métodos más conocidos y aplicados en la industria; son útiles en ausencia de datos empíricos cuantificados. Estas técnicas permiten interpretar y acumular el conocimiento y la experiencia de los profesionales, generalmente los gerentes de proyecto; las estimaciones dadas por los expertos son una síntesis de los conocimientos prácticos adquiridos en proyectos anteriores [18]. La principal desventaja de este método es la alta dependencia del experto que se genera para realizar la estimación; de otra parte no se cuenta con mecanismos que permitan evaluar qué tan buena ha sido la estimación. hasta que empíricamente se comprueba que los resultados no la avalan; descubrimiento que ya resulta demasiado tardío para corregir los daños producidos por la incorrecta opinión concebida [18]. A continuación se describen los principales métodos de estimación en esta categoría.

Método de estimación por analogía

La estimación por analogía puede servir para empresas que estén comenzando a crecer rápidamente. Las horas de mano de obra directa necesarias para fabricar un componente pueden estimarse recurriendo a las horas que fueron necesarias para trabajos similares. La base para la estimaciones la similitud que existe entre el elemento conocido y la pieza propuesta. Algunos estimadores con experiencia, tales como mecánicos, fabricantes de herramientas o técnicos, pueden estimarlos tiempos necesarios con mucha precisión. Por tanto, se les suele consultar cuando se necesita una estimación rápida. En la encuesta se consideró el método de estimación por analogía equivalente al método de estimación basado en componentes genéricos.

En todos los niveles, muchas estimaciones se realizan por analogía. Por ejemplo, el proyecto A necesitó 10.000 horas de mano de obra directa y 4.000 horas de equipos. Dadas las similitudes y diferencias entre el proyecto A y el proyecto B propuesto, el número de horas de mano de obra directa y de horas de equipos pueden estimarse en 8.000 y 3.200 respectivamente. Aplicando la tarifa actual de las horas de mano de obra y las de equipos, y una tarifa global de gastos generales, puede estimarse un coste total del proyecto. O bien el estimador puede encontrar elementos del proyecto A que sean análogos a elementos del proyecto B, de lo que puede estimarse el coste del proyecto B. En este ejemplo, la analogía pasa a formar parte del método de estimación de ingeniería, [26].

Un inconveniente importante de la estimación por analogía es el alto grado de conocimientos exigido. Se necesita una considerable experiencia y pericia para identificar y usar las analogías apropiadas y para hacer ajustes en las diferencias percibidas. Sin embargo, como el coste de la estimación por analogía es bajo, puede usarse como comprobación de otros métodos. A menudo es el único método que puede utilizarse, porque el producto, sistema o servicio está sólo en una etapa preliminar del desarrollo, [26].

Estructura de la Descomposición de Trabajo (EDT).

También conocido como Work Breakdown Structure (WBS), es el método clásico usado en la ingeniería de software y la gerencia de proyectos para descomponer un proyecto en pequeñas actividades medibles. En estas estructuras se definen los elementos de trabajo tangibles de un proyecto de una manera que ayuda a organizar y definir el alcance total del proyecto, propuesto y costo. Esta estimación por desglose del trabajo está ampliamente apoyada por el PMBOK [52], y el SWEBOK [63], en los capítulos de planificación de proyectos.

Una EDT es una presentación simple y organizada del trabajo requerido para completar el proyecto, existiendo muchas maneras de organizar la presentación de este trabajo. Sin

embargo, para que esta herramienta sea verdaderamente útil se debe atender a que su característica fundamental es ser orientada a los entregables o "productos del trabajo" que son el resultado del esfuerzo y no el esfuerzo en sí. De esta manera, para construir una EDT se debe tener claridad respecto del alcance del trabajo a ejecutar en el momento de elaborar la estimación, el que debe estar documentado en la "Declaración de Alcance del Proyecto". El proceso de desglose o descomposición debe ser progresivo y representar siempre el alcance completo, esto significa que para generar la EDT se debe proceder desde lo general a lo particular, y cada nivel debe ser el resultado de la integración del nivel siguiente. Algo importante de recordar es que la EDT documenta el alcance del proyecto, no su plan de ejecución. [21], [52]

Una buena práctica es que la EDT sea lo suficientemente detallada como para poder asignar una parte del trabajo a un tercero, y que su estado se monitorice adecuadamente. Por eso una buena medida práctica de la profundidad que debe alcanzar la EDT es preguntarse si el nivel al que se ha llegado permite estimar con claridad las variables tiempo y costo. Si aún no se pueden determinar en el nivel en que se está, deberá subdividirse aún más. Este trabajo asignado podría convertirse en una nueva EDT subalterna de la anterior. Generalmente una EDT no debe tener más de 100 o 200 elementos terminales (si parece que se requieren más, se deberían utilizar sub proyectos).

Una EDT debería tener 3 o 4 niveles de profundidad, y cada nivel debería tener entre 5 y 9 elementos de ancho. Estas sugerencias se derivan de los siguientes hechos:

- La capacidad de memoria a corto plazo está restringida a entre 5 y 9 elementos.
- Teniendo un tiempo fijo para planear el proyecto, a mayor cantidad de elementos terminales, habrá menor tiempo para prestar atención a cada uno de ellos. En consecuencia, los estimados son menos pensados.
- Es conveniente y bastante común la práctica de usar, en proyectos medianos y grandes, un sistema de código jerárquico, asignando un código a cada entrada de la EDT. Por ejemplo, una entrada del nivel más alto puede tener un código como 1, 2 ó 3, y las entradas bajo la entrada 1 pueden tener códigos como 1.1, 1.2, 1.3, etc. [21], [52].

Método Delphi y Delphi Banda Ancha

Es una técnica basada en el consenso para la estimación de esfuerzo. Se deriva del método Delphi, que se desarrolló en el 1950-1960 en la RAND Corporation, como una herramienta de previsión. Desde entonces, ha sido adaptado en muchas industrias para estimar muchos tipos de tareas, que van desde los resultados estadísticos de recolección de datos de previsiones de ventas y marketing. Está basado en juicios de expertos considerando las respuestas a un cuestionario. En el área de estimación de software se toman las consideraciones que tuvieron los expertos para estimar (en forma de evaluaciones cuantitativas y comentarios

escritos) y estos son provistos como retroalimentación a los mismos expertos como partes de una ronda siguiente del cuestionario (next-round). A continuación, los expertos revalúan sus opiniones a la luz de esta información, y se genera un consenso de grupo con respecto a la estimación. [56], [69].

Según [56], para realizar una buena estimación es necesario seguir ciertos pasos como:

- Explicar a los expertos en qué consiste el método. Con esto se pretende conseguir la obtención de previsiones fiables, pues los expertos van a conocer en todo momento cuál es el objetivo de cada una de los procesos que requiere la metodología.
- Delimitar el contexto en el que se desea realizar la previsión sobre el tema en estudio
- Seleccionar el panel de expertos y conseguir su compromiso de colaboración. Las personas que sean elegidas deben presentar una pluralidad en sus planteamientos, es decir, deben tener diferentes perspectivas del proyecto (arquitectura, requisitos, tecnología, etc.) para que de esta manera se garantice un análisis más integral.

Métodos ágiles de estimación.

Son técnicas de estimación que están basadas en los principios del método Delphi de Banda Ancha, en general comparten dos características fundamentales: dar agilidad al proceso de estimación e involucrar a todo el equipo en la creación de estimaciones creíbles y compatibles. Los siguientes son algunos ejemplos de técnicas “ágiles” de estimación:

- **Planning Poker:** es una herramienta para la estimación de los proyectos de desarrollo de software. En esta técnica se le pide a cada miembro del equipo para que escoja su tarjeta de estimación de tal manera que no puede ser visto por los demás jugadores. Después de que cada jugador ha seleccionado una tarjeta, todas las tarjetas son expuestas a la vez. [51], [67]
- **Estimación orientada a pequeñas iteraciones.** El más conocido en este sentido es la estimación que se realiza dentro del método ágil SCRUM de desarrollo de software, de cada una de las iteraciones denominadas Sprint; un Sprint es un lapso de tiempo generalmente entre una semana y un mes, denominado “timeboxed” (es decir, están limitados a una duración determinada) con un esfuerzo constante. El propósito de la planificación enmarcada en un Sprint es decidir sobre los compromisos y asegurar su contenido sea claramente comunicado.

El equipo necesita entonces proporcionar una estimación rápida de cada elemento del producto de trabajo en orden de prioridad, en base a esta prioridad, se dejan por fuera otras actividades que no necesitan ser discutidas durante el lapso de tiempo. [57], [61].

2.3. Conceptos básicos de la Ingeniería de Software Experimental

2.3.1. Importancia y retos de la experimentación en Ingeniería de Software

Durante mucho tiempo los científicos se han comprometido a describir y organizar la información obtenida por las observaciones de campo. Las ideas sobre la experimentación se han explorado y desarrollado en diferentes áreas científicas, incluyendo la física, la agricultura, la medicina, la ingeniería y las ciencias sociales, entre otros.

A pesar de la madurez de los estudios experimentales en otros campos de conocimiento, la aplicación de estos enfoques en el campo de la ingeniería de software esta aun en pleno desarrollo. Victor R. Basili en [7] , hace un recorrido de 40 años de la evolución de la ingeniería de software experimental, comenzando desde el pasado hacia el futuro. Una de las principales razones de realizar investigaciones empíricas en el campo de ingeniería de software es la necesidad de conseguir evidencia objetiva, soportada en resultados significativos estadísticamente, con respecto a la adopción de nuevos métodos o tecnologías en el desarrollo de software [7] .

Durante las últimas dos décadas las investigaciones en ingeniería de software ha alcanzado considerables resultados para entender la aplicación y la evolución de los procesos de software y tecnologías mediante la aplicación del método científico, usando experimentación para apoyar sus investigaciones.

Es por esto que hoy en día, la experimentación ha demostrado ser una necesidad para la evolución del campo. De hecho, los enfoques basados en la experimentación ya representan una herramienta cardinal para permitir la transferencia de tecnologías de software para la industria, mejorar los procesos de software y los comportamientos de prueba en Ingeniería de Software. [5], [37], [37], [58], [64], [70]. Como lo menciona V. R. Basili en [7] , la Ingeniería de Software Experimental “Es una Gran Ciencia”.

Como paradigmas de investigación se pueden mencionar los siguientes, según [64]:

- **Paradigma Analítico:** el cual propone una teoría formal, por medio de una serie de axiomas, desarrolla la teoría, obtiene los resultados y cuando es posible, verifica los resultados con observaciones empíricas.
- **Paradigma experimental:** el cual se ocupa de observar soluciones existentes, proponer un modelo, una teoría acerca del comportamiento, o una mejor solución a la actual, medir ya sea por usando variables cuantitativas o cualitativas, analizar los datos recolectados y validar el proceso o producto bajo estudio.

A pesar de las ventajas evidentes de la aplicación de experimentación en el campo de la ingeniería de software, en el campo aplicado, por lo menos en la industria de software colombiana, la adopción de este enfoque se encuentra muy incipiente. Cada empresa adopta modelos, métodos y técnicas sin una validación formal, simplemente por no quedarse atrás en la adopción tecnológica, o siguiendo la tendencia del mercado y sin tener evidencia objetiva de su impacto.

2.3.2. Dificultades de la experimentación.

De acuerdo con [64], la dificultad de realizar estudios experimentales en el área de Ingeniería de Software radica en los siguientes elementos:

- La naturaleza la Ingeniería de Software trata sobre un desarrollo a la medida y no es una producción en cadena como en otras disciplinas, por ejemplo manufactura, una fábrica.
- Este proceso está basado en humanos y no en maquinas, por lo cual los productos son más propensos a tener errores.
- No todos los software son iguales, existen un gran número de variables que causan diferentes efectos y todos estos efectos necesitan ser entendidos.
- Actualmente existe una necesidad de modelos que ayuden a razonar sobre esta disciplina.
- No existe una clara definición de los límites de la tecnología en ciertos contextos.
- Existe un análisis insuficiente y una pobre experimentación.

2.3.3. Elementos de la experimentación

Según [67], los elementos básicos de un experimento son los siguientes:

- **Las variables:** Las cuales pueden definirse en dos tipos, variables dependientes e independientes. Las variables independientes se refieren a las entradas del proceso de experimentación, estas variables también son conocidas como “factores” y representan la causa que afecta el resultado del proceso de experimentación. El valor propio que toma un factor se denomina “tratamiento”. Las variables independientes se refieren a la salida del proceso de experimentación. Estas variables representan el efecto que es causado por los factores del experimento. El valor que toma una variable dependiente se denomina “resultado”.

- **Los objetos:** son una herramienta usada para verificar la relación causa-efecto en una teoría. Durante la ejecución del experimento los tratamientos están siendo aplicados a un conjunto de objetos y así el resultado está siendo evaluado. Los objetos junto con el sistema de medición y las directrices de ejecución del experimento componen la instrumentación del experimento. El Anexo A, presenta la instrumentación de nuestro estudio experimental.
- **Los participantes:** son los individuos que fueron seleccionados de la población de interés para conducir el experimento. Esto significa que para generalizar los resultados de un experimento a una población deseada, el conjunto de participantes debe ser representativo para aquella población. Para alcanzar esto, los parámetros de selección de participantes y tamaño del conjunto seleccionado, deben ser considerados. En principio, cuanto más variable sea la población tanto mayor debe ser el tamaño del conjunto de participantes.
- **El contexto del experimento:** está compuesto de las condiciones en que el experimento está siendo ejecutado. El contexto puede ser caracterizado de acuerdo a las siguientes dimensiones:
 - a. In vivo/ in vitro / in virtuo / in silico
 - In vivo:** Considera el estudio de un proyecto real. No se necesitan modelos.
 - In vitro:** Se refiere a la experimentación en el laboratorio teniendo las condiciones controladas. Se utilizan un ambiente modelado.
 - In virtuo:** El objeto y el ambiente son modelados por computador.
 - In silico:** El comportamiento del sujeto, el objeto y el ambiente son todos modelados por computador.
 - b. Alumnos / Profesionales

Define el tipo de personas que van a ejecutar el experimento
 - c. Problema de sala de aula / Problema real

Muestra el tamaño y la complejidad del problema que está siendo estudiado.
 - d. Especifico / General

Muestra si los resultados del experimento son validos para un contexto partículas o para el dominio de la Ingeniería de Software como tal.
- **Las hipótesis:** Un experimento es formulado generalmente a través de hipótesis. La hipótesis principal se llama hipótesis nula y declara que no existe ninguna relación estadísticamente significativa entre la causa y el efecto. El objetivo principal del experimento es entonces, rechazar la hipótesis nula, a favor de una o algunas hipótesis alternativas. La decisión sobre rechazo de la hipótesis nula puede ser tomada en base

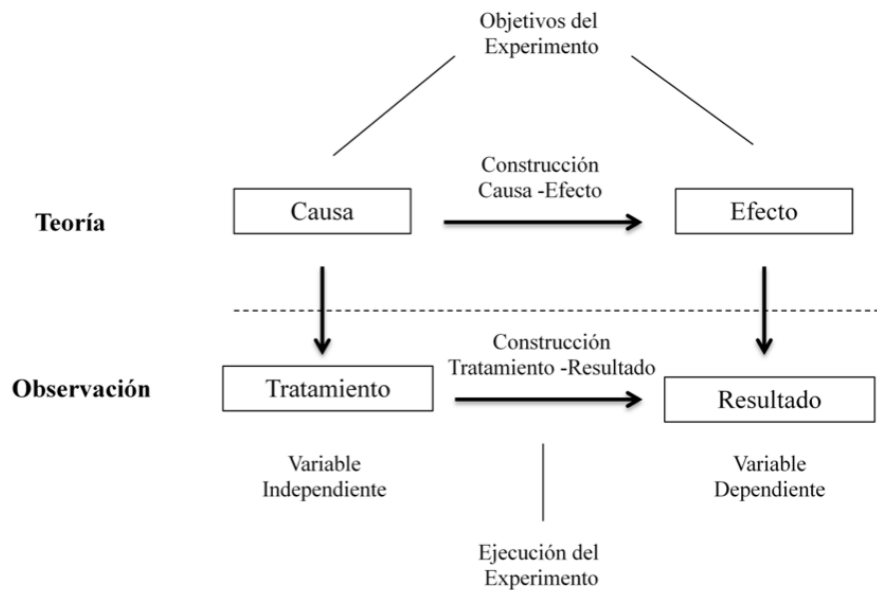


Figura 2.4: Los conceptos de un experimento. Tomado de [67], [70].

a los resultados y de su verificación utilizando una prueba estadística.

- **El proyecto del experimento:** determina la manera como el experimento será conducido. La decisión sobre asignación de los objetos y de los participantes es realizada en este momento. También la manera como los tratamientos serán aplicados a los objetos es definida. La combinación de los objetos, participantes y tratamientos se llama “prueba experimental” o “trial”. La cantidad y la secuencia de estas pruebas piloto experimentales se definen en el proyecto del experimento. Información más detallada respecto al proyecto del experimento puede ser encontrada en [37].

La Figura 2.4, presenta la relación entre los conceptos descritos anteriormente. Tomado de [67], [70].

2.3.4. Definición del objetivo del experimento

Basados en la metodología Goal/Question/Metrics descrita en [53], se propone el uso de cuatro elementos para construir el objetivo del estudio experimental, los cuales son:

- **Objeto de estudio:** puede ser un proceso, un producto o un modelo formal.

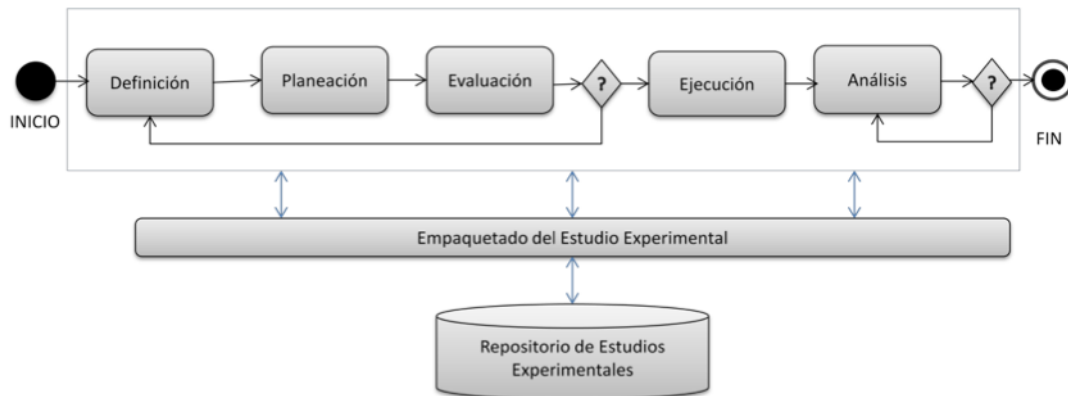


Figura 2.5: Proceso del Estudio Experimental. Tomado de [64], [65]

- **Propósito:** El propósito del estudio puede ser
 Caracterizar, que responde a la pregunta: ¿Que está pasando en la realidad? Evaluar,
 ¿Es correcto, es de buena calidad?
 Controlar: ¿Puedo controlar eventos?
 Mejorar: ¿Se pueden mejorar estos eventos?
 Predecir: ¿Qué se puede esperar para el futuro?
- **Foco del estudio:** es el objeto de interés en el estudio.
- **Punto de vista:** Los beneficiarios de la investigación.

2.3.5. Proceso del estudio experimental

En este trabajo se sigue la propuesta definida en [64], [65] donde se proponen los siguientes pasos del proceso básico de un estudio experimental que se pueden observar en la Figura 2.5.

2.4. Trabajos experimentales relacionados a la estimación de software

Existen en la literatura estudios experimentales orientados hacia la estimación de proyectos de software; la Tabla 2.5, sintetiza estudios experimentales recientes que guardan alguna similitud con esta investigación [11], [17], [43], [45], [62], [65]. Algunos de estos trabajos

son realizados en grupos de empresas de diversos países como Noruega [43], China [17], Pakistán [45] y Brasil [65]; este último trabajo [65] hace referencia a un perfil de las empresas brasileñas que han adoptado el modelo MPS.BR de mejora de la calidad de software el cual es un referente importante para esta investigación.

Con respecto a la precisión de la estimación del esfuerzo y la duración de proyectos de software, se encontraron estudios que han sido desarrollados en diferentes países desde 1984 y que fueron recopilados por Da Yang et. al. [19], el cual analiza el desfase de la estimación tanto del esfuerzo como de la calendarización. Ver Tabla 2.6.

En general, se observa en estos estudios que el desfase del esfuerzo es mucho mayor que el desfase en la calendarización de los proyectos. Es decir, es común en la industria de software la preocupación por cumplir a toda costa con el plazo de entrega y para lograrlo se requiere un sobre esfuerzo en los integrantes del proyecto o se asigna más personal a proyectos atrasados, lo cual es contraproducente. Una buena estimación involucra entonces entregar todas las funcionalidades en la fecha indicada y con los niveles de esfuerzo estimados.

Si bien los valores de desfase de la estimación del esfuerzo y la calendarización presentados en [19] y [66] no pueden ser comparados puesto que se llevaron a cabo en contextos y condiciones diferentes, será interesante analizar si en la muestra de empresas objeto de estudio se sigue presentando esta tendencia.

Referencia	Propósito	Resultados y limitaciones
[43] (Molokken et al. 2004)	Encuesta a 18 compañías diferentes y 52 gerentes de proyectos acerca la práctica de estimación.	El promedio del esfuerzo real sobrepasa en un 41 % al esfuerzo estimado. Estimación basada en juicio de expertos es el método de estimación más usado.
[62] (Subramanian et al. 2006)	Estudio acerca del efecto de la complejidad, plataforma y lenguaje de programación en el esfuerzo de desarrollo de Sistemas de Información.	Se usaron datos de 666 programas de 15 proyectos de software diferentes, se encontró que las tres variables tienen un efecto significativo en el esfuerzo. Programas por lotes exigen un mayor esfuerzo de desarrollo que los sistemas on-line.
[19] (Da Yang et al. 2008)	Identificar posibles áreas de mejora en la estimación de software, analiza 112 proyectos y realiza 116 encuestas a empresas.	Proyectos grandes son más propensos a sobre esfuerzo y demora en la entrega. Se tomo el banco de datos Chinese Software Benchmarking Standards Group (CSBSG). Se investigaron las posibles causas de la imprecisión en las estimaciones.
[45] (Nasir et al. 2008)	Enfocado en la brecha que existe entre las empresas de software respecto a CMMI-DEV que sugiere la adopción de métodos formales para estimar	Propone una categorización de técnicas de estimación y las divide en paramétricas y heurísticas. Estudio de 4 proyectos de software de una sola compañía desarrolladora.
[11] (Capretz, et al. 2009)	Propone un nuevo método de estimación llamado Voting framework	Se tomo el banco de datos: International Software Benchmarking Standards Group (ISBSG) [47]. Se tomaron 439 proyectos de software y se realizo la estimación con 5 métodos paramétricos.
[66] (Travassos et al. 2011)	iMPS 2010 : Desempeño de las empresas que adoptaron el modelo MPS de 2008 a 2010.	Encuesta anual realizada a empresas que adoptaron MPS.BR y que permite analizar diversos aspectos entre los cuales se encuentra la precisión de las estimaciones.

Tabla 2.5: Síntesis de Trabajos Relacionados

Referencias	[35]	[50]	[28]	[39]	[8]	[55]	[43]	[19]	[66]
Año	1984	1988	1989	1991	1992	2003	2004	2007	2010
Desfase del es- fuerzo	34 % Me- diana	33 % Me- dia			33 % Me- dia	18 % Me- dia	21 % Me- diana 41 % Me- dia	5 % Me- diana 12 % Me- dia	
Esfuerzo Actual>Estimado	61 % Me- dia		70 % Me- dia	63 % Me- dia		59 % Me- dia	76 % Me- dia	68 % Me- dia	
Esfuerzo Actual<Estimado	10 % Me- dia			14 % Me- dia		15 % Me- dia	19 % Me- dia	29 % Me- dia	
Desfase de la Ca- lendarización	22 % Me- dia					23 % Me- dia	9 % Me- diana 25 % Me- dia	7 % Me- diana 17 % Me- dia	33 % Me- dia
Calendarización Actual>Estimado	65 % Me- dia		80 % Me- dia			35 % Me- dia	62 % Me- dia	63 % Me- dia	
Calendarización Actual<Estimado	4 % Me- dia					3 % Me- dia	2 % Me- dia	21 % Me- dia	

Tabla 2.6: Comparación de la precisión de la estimación. Adaptado de [19].

Capítulo 3

Definición del Estudio Experimental en Estimación de Software

En este capítulo se describen las fases del diseño y la planeación del estudio experimental, tomando como referencia la metodología aplicada en el Grupo de Ingeniería de Software Experimental, PESC-ESE/COPPE/UFRJ, de la Universidad Federal de Rio de Janeiro [64], [65], [67]. Se realizó la identificación de las características generales del estudio, definición de los objetivos y preguntas de investigación, la formulación de las hipótesis, definición de las variables involucradas, identificación de los recursos requeridos y la manera cómo fueron analizados los datos y como fue verificada la validez del estudio.

3.1. Caracterización del estudio

3.1.1. Tipo o de estudio Encuesta.

In-vivo

3.1.2. Dominio.

Ingeniería de Software, Gestión de Proyectos de Software, Estimación de software.

3.1.3. Idioma.

Español

3.1.4. Participantes

- Universidad EAFIT. Departamento de Informática y Sistemas. Grupo de I+D+I Investigación en TIC, línea de Ingeniería de Software.
<http://www.eafit.edu.co/investigacion/grupos/escuela-ingenieria/i-d-i-tic/Paginas/i-d-i-tic.aspx>
- Red Colombiana de Calidad para el Desarrollo en Tecnologías de Información y Comunicación RCC- TIC. <http://www.rcctic.org/> [54]. A través de esta entidad se hizo contacto con las empresas que participaron en el estudio.
- UFRJ/COPPE/ PESC-ESE Grupo de Investigación en Ingeniería de Software Experimental. En donde el autor de este trabajo realizó una estancia de 3 meses de investigación “Short Stay Research” patrocinada por LACCIR (<http://www.laccir.org/>), bajo el código S1110LAC004. Durante esta estadía se recibió la asesoría del doctor Guilherme H. Travassos como Asesor Metodológico del Estudio Experimental.
<http://www.cos.ufrj.br/>

3.1.5. Número de repeticiones estimadas.

Se planeo hacer una sola ejecución en las ciudades de Bogotá, Medellín, Cali y Bucaramanga, teniendo en cuenta que son las principales ciudades en las que se concentran las empresas que participaron en el proyecto de acompañamiento masivo. Por otra parte, dado que la estrategia de experimentación escogida en este estudio fue de tipo encuesta (Survey) se tiene una alta facilidad de repetición del experimento a un bajo costo.

3.2. Definición del estudio experimental

3.2.1. Objetivo General

Analizar: La estimación de proyectos de software.

Con propósito de: Caracterizar.

Con respecto a: La utilización de las prácticas de estimación descritas en el modelo CMMI-DEV v1.2.

Desde el punto de vista de: Los gerentes de proyectos de software.

En el contexto de: Las compañías de software Colombianas que participaron en el proyecto masivo para implantación del CMMI-DEV v1.2.

3.2.2. Objetivos Específicos

Con base en el proceso de estimación.

- Determinar cuáles son las prácticas actuales de estimación en las empresas.
- Determinar cuál es el nivel de aplicación de las prácticas de estimación planteadas en el modelo CMMI-DEV v1.2 en las empresas.
- Determinar la precisión de la estimación del esfuerzo y la precisión de la estimación de la duración.
- Determinar cuáles son las oportunidades de mejora en la adopción de nuevas prácticas de estimación de software según los gerentes de proyectos.

El análisis de prácticas relacionadas con estimación pueden ser agrupadas en diferentes categorías; este análisis de prácticas desde la perspectiva de conjuntos es una adaptación del modelo presentado en [67], y sirvió de punto de partida para definir las premisas que fueron tomadas en consideración por cada objetivo específico, ver Figura 3.1.

Tomando como referencia el conjunto de prácticas recomendadas por el modelo CMMI-DEV v1.2, ver Tabla 3.1, se identifican dos subconjuntos de prácticas desde la perspectiva de las empresas: prácticas aplicables y prácticas no aplicables. Las prácticas aplicables se pueden subdividir a la vez en prácticas que ya están institucionalizadas en la organización y prácticas que están en proceso de adopción.

Con base en las metodologías de estimación.

- Identificar cuáles son los métodos de estimación más utilizados en las empresas, ver Tabla 3.2.
- Identificar como son aplicados dichos métodos: por tipo de proyecto, a manera general de toda la organización o se utiliza más de un método de estimación por proyecto.

3.2.3. Foco de calidad.

El foco de calidad va a ser visto desde dos dimensiones:

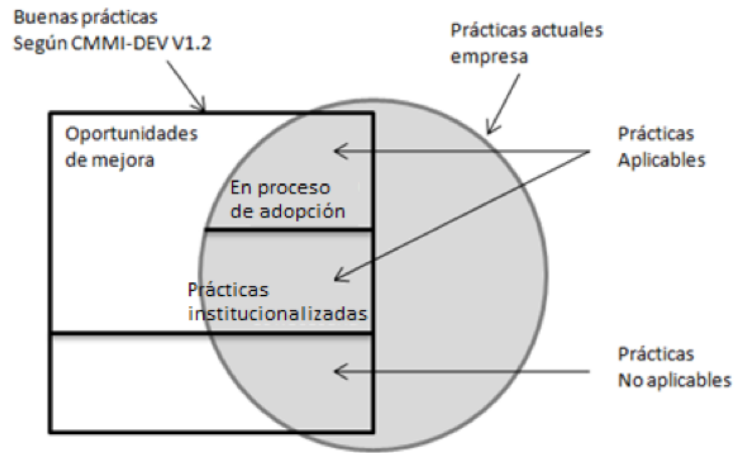


Figura 3.1: Modelo conceptual de la encuesta. Adaptado de [67].

Desde el proceso de estimación:

El foco de calidad del proceso consistió en el nivel de cubrimiento que la compañía tuvo respecto a las prácticas reconocidas por el modelo de calidad. En la Tabla 3.1, se identifican las prácticas específicas, mientras que en la Tabla 3.2 se identifican las prácticas genéricas sugeridas por CMMI-DEV. v1.2, que sirvieron como referente para evaluar el nivel de aplicabilidad y apropiación en la organización.

El modelo CMMI-DEV distingue las prácticas en dos categorías: Prácticas específicas que aplican a un área de proceso en particular y prácticas genéricas que representan guías que apoyan la institucionalización de los procesos en una organización. Las prácticas específicas relacionadas con la estimación se encuentran inmersas en las actividades de planificación y monitoreo y control del proyecto y que se corresponden con las áreas de proceso de PP (Project Planning) y PMC (Project Monitoring and Control)., respectivamente. Se identificaron 5 prácticas específicas directamente relacionadas con la estimación.

Teniendo en cuenta que las empresas a las cuales se dirigió el estudio, las cuales están iniciando la adopción de CMMI, se tomaron con punto de referencia las 10 prácticas genéricas asociadas a la meta genérica de un Proceso Gestionado y que desde la perspectiva del modelo, se corresponde con el nivel de madurez 2 (GG 2 Managed Process).

Área de Pro- ceso (PAs)	Metas Es- pecíficas (SG) y Practicas Específicas (SP)	Interpretación de la Practicas Específicas en el contexto de la Estimación	Id. Prácti- ca
Project Monitoring and Control (PMC)	SG 1 SP 1.1	Monitorear los valores actuales de los parámetros de planeación del proyecto, tales como esfuerzo y duración del proyecto, comparándolos con los valores estimados en el plan de proyecto e identificar desviaciones significativas.	P1
	SG 2 SP 2.2	En caso de existir una desviación significativa en la ejecución del proyecto de acuerdo a lo planeado se ejecutaran las acciones correctivas determinadas en el Plan del Proyecto.	P2
Project Planning (PP)	SG 1 SP 1.1	Establecer una estructura de ítems de trabajo a un nivel general para estimar el alcance del proyecto.	P3
	SG 1 SP 1.2	Establecer y mantener los estimativos para las tareas y los ítems de trabajo.	P4
	SG 1 SP 1.4	Estimar el esfuerzo de los productos de trabajo basados en una estimación a conciencia.	P5

Tabla 3.1: Prácticas específicas de estimación sugeridas por CMMI-DEV v1.2

Prácticas Genéricas (GP)	Interpretación de la Practicas Genéricas en el contexto de la Estimación	Id. Práctica
GP 2.1	Establecer y mantener una política organizacional para la planeación y el desempeño del proceso de estimación de proyectos de software.	P6
GP 2.2	Establecer y mantener un tiempo requerido para realizar la estimación en la planificación de los proyectos.	P7
GP 2.3	Proveer los recursos adecuados para desarrollar las actividades de estimación planeadas, tales como: estimadores expertos, datos históricos, productos de trabajo, listas de tareas, etc.	P8
GP 2.4	Asignar autoridad y responsabilidad a los estimadores e involucrar a todos los participantes del proyecto.	P9
GP 2.5	Entrenar a los participantes del proyecto en la metodología de estimación, modelos, datos y/o algoritmos usados para determinar la estimación.	P10
GP 2.7	Identificar e involucrar en la estimación a las personas responsables de ejecutar las tareas.	P11
GP 2.8	Monitorear y controlar el progreso real de los proyectos contra lo estimado y establecer criterios de acción correctiva en caso de una desviación importante de los proyectos a nivel organizacional.	P12
GP 2.9	Evaluar objetivamente la adherencia del proceso de estimación usando modelos paramétricos de estimación o modelos de juicio de expertos basados en datos históricos y controlar posibles adaptaciones del proceso.	P13
GP 2.10	Revisar el estado de las actividades y los resultados del proceso de estimación con la alta gerencia y determinar cuál es su nivel de satisfacción con respecto a la manera en que se está llevando a cabo la estimación en la organización.	P14

Tabla 3.2: Prácticas genéricas de estimación sugeridas por CMMI-DEV v1.2

Identificador	Método de Estimación
JEX	Juicio de Expertos
DDB	Delphi, Delphi Banda Ancha
PCU	Puntos de Casos de Uso
PFA	Puntos de Función Ajustados
BCG	Basado en componentes genéricos
PPK	Planning Poker
CCO	COCOMO, COCOMO II
BRN	Modelo Basado en Redes Neuronales
FLO	Lógica Difusa (Fuzzy Logic)
EST	Otros modelos estadísticos
MP	Modelo propio de la empresa

Tabla 3.3: Métodos de estimación seleccionados para el estudio. Adaptado de [45].

Desde los métodos de estimación:

El criterio de selección de los métodos que formaron parte del estudio se basó en la experiencia de los asesores, y tomando como referencia otros estudios como [45], los cuales proponen una lista de métodos de estimación comúnmente usados, el cual los clasifica en métodos paramétricos y métodos heurísticos y se adicionaron los Métodos de Medición del Tamaño Funcional del Software, basados en la norma ISO (IEEE ISO/IEC 14143- 1:2007) [30], entendiendo que son métodos que tienen algún nivel de adopción en Colombia, ver Tabla 3.3, lista de métodos de estimación seleccionados para el estudio.

3.3. Perspectiva de observación

Desde la perspectiva de observación de los gerentes de proyectos que generalmente son los responsables de la estimación y monitoreo de proyectos en las Pymes de software.

3.4. Selección del contexto

El contexto del estudio experimental puede ser analizado desde cuatro dimensiones:

El proceso:

Se trata de un estudio in-vivo, puesto que la encuesta se aplicó a las empresas participantes, donde los gerentes de proyecto respondieron en base a su propia experiencia.

Los participantes:

La encuesta fue diligenciada por profesionales en el área de la ingeniería de software con experiencia en el área de gerencia y estimación de proyectos.

La realidad:

Se trata de un problema real por que se están tomando los datos proporcionados por las empresas.

La generalidad:

El contexto del estudio posee un carácter específico ya que se enfoca a las empresas de desarrollo de software que hicieron un ciclo de mejora bajo un mismo proyecto de acompañamiento masivo, tomando como referente un mismo modelo de calidad.

3.5. Preguntas de investigación y métricas asociadas a estas

Las preguntas de investigación y las métricas asociadas a cada pregunta, son las siguientes: Las preguntas RQ1, RQ2 y RQ7 están basadas en una variable cualitativa, denominada Nivel de Aplicación; Por cada una de las prácticas de estimación descritas en las Tabla 3.1 y Tabla 3.2, los participantes seleccionaron una opción entre las siguientes:

- a. Practica no aplicable en la empresa
- b. Practica no utilizada en la empresa
- c. Practica utilizada pocas veces
- d. Practica utilizada en algunos proyectos
- e. Practica institucionalizada en todos los proyectos.

RQ1: ¿Cuáles son las prácticas de estimación más aplicadas en las empresas?

M1: Conteo simple de basado en un métrica cualitativa; prácticas con mayor número de ocurrencias de respuestas en la categoría (e).

Ejemplo: Se identificó la práctica P2 como aquella con un mayor número de respuestas de valor (e).

RQ2: ¿Cuáles es el nivel de aplicación de las prácticas consideradas útiles para la organización?

M2: Conteo simple de número de ocurrencias de respuestas en cada una de las categorías tipo (c), tipo (d) y tipo (e).

Ejemplo: Para la práctica de estimación P2, se identifico el número de respuestas tipo (c), tipo (d) y tipo (e).

RQ3: ¿Cual es la precisión de estimación del esfuerzo en los proyectos de software?

M3: Métrica cuantitativa. Esta dada en unidades horas/hombre.

Ejemplo: La empresa ABC reportó un esfuerzo promedio estimado de XX horas/hombre y un valor real del esfuerzo de YY horas/hombre, para los proyectos concluidos en el 2010. En la sección 4.3.3 se explica en detalle la formula que se ejecuto para calcular la precisión de la estimación del esfuerzo.

RQ4: ¿Cual es la precisión de estimación de la duración en los proyectos de software?

M3: Métrica cuantitativa. Esta dada en unidades de semanas.

Ejemplo: La empresa ABC reportó una duracio?n promedio estimada de XX semanas y un valor real de la duración de YY semanas, para los proyectos concluidos en el 2010. En la sección 4.3.4 se explica en detalle la formula que se ejecuto para calcular la precisión de la estimación de la duración.

RQ5: ¿Cuales son las oportunidades de mejora y factores de éxito que perciben las compañías en el proceso de estimación?

M5: Métrica cualitativa. Pregunta abierta.

Ejemplo: Cada participante expreso libremente las oportunidades de mejora y factores de éxito que considera encuentra dentro de su organización. Se busco por respuestas que coincidieran entre las diferentes empresas.

RQ6: ¿Cuál es el nivel de aplicación en la organización de los métodos de estimación reconocidos en la literatura?

M6: Métrica cualitativa, conteo simple de número de ocurrencias del mismo valor. Tomando como base el listado de métodos de estimación propuesto en la Tabla 3.3. Los participantes seleccionaron por cada metodología el nivel de aplicación de 1 a 5, dentro de la empresa, donde:

- (1) No lo conozco
- (2) Lo conozco en teoría
- (3) Aplicado a un proyecto piloto
- (4) Aplicado a criterio del gerente de proyectos
- (5) Institucionalizado y calibrado de acuerdo al contexto de la organización

Ejemplo: Por cada método de estimación, se identifico el número de respuestas tipo (1), tipo (2), tipo (3), tipo (4) y tipo (5), para identificar el nivel de aplicación.

RQ7 ¿De qué manera se aplican en la organización los métodos de estimación?

M7: Métrica cualitativa, conteo simple de número de ocurrencias del mismo valor. Se aplica la misma escala de Nivel de aplicación para responder dos preguntas puntuales:

A. ¿Utiliza más de un método de estimación para el mismo proyecto, con el propósito de disminuir el nivel de incertidumbre de la estimación?

B. ¿Utiliza distintos métodos de estimación para los diferentes tipos de proyecto?

Ejemplo: Por cada pregunta, se identifica o el número de ocurrencias de cada respuesta.

RQ8 ¿Cuál es el nivel de satisfacción de los representantes de la organización con respecto a la estimación?

M8: Métrica cualitativa, conteo simple de número de ocurrencias del mismo valor. Los participantes seleccionaron una opción de la siguiente lista de opciones:

- a) Totalmente satisfecho
- b) Satisfecho
- c) Parcialmente satisfecho
- d) Insatisfecho
- e) Totalmente insatisfecho

Ejemplo: Se identifica el número de ocurrencias de cada respuesta y se saca el promedio.

3.6. Planeación

3.6.1. Formulación de hipótesis

Para la construcción de las hipótesis, partimos de la construcción de la hipótesis nula la cual es una hipótesis construida para anular o refutar con el objetivo de apoyar una hipótesis alternativa. La hipótesis nula se presume verdadera hasta que una prueba estadística en la forma de una prueba empírica pruebe lo contrario. Si la hipótesis nula no es rechazada, esto no quiere decir que sea verdadera.[29]

Vista del proceso de estimación

Hipótesis Nula: (H0): Todas las prácticas de estimación recomendadas por el modelo CMMI-DEV v.1.2 son aplicables a todas las organizaciones.

A: prácticas actuales de estimación en las empresas

B: prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2.

$$H0 : B-(A \cap B) = \emptyset$$

Hipótesis Alternativa (H1): Existe un conjunto de prácticas recomendadas por el modelo CMMI-DEV v1.2, que son aplicables en las organizaciones.

A: prácticas actuales de estimación de las empresas participantes del estudio.

B: prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2.

$$H1 : B-(A \cap B) \neq \emptyset$$

Hipótesis Alternativa (H2): Dentro de las prácticas de estimación de las empresas participantes del estudio, que hacen parte de las prácticas planteadas en el modelo de calidad CMMI-DEV v1.2, existen algunas prácticas de estimación que los gerentes de proyecto consideran no aplicables.

C: prácticas actuales de estimación de las empresas participantes del estudio que hacen parte de las prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2.

$$C = A \cap B.$$

Cu: prácticas actuales de estimación de las empresas participantes del estudio, que hacen parte de las prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2 y que los gerentes de proyecto consideran no aplicables.

$$H2 : C-(C \cap Cu) \neq \emptyset$$

Hipótesis Alternativa (H3): Dentro de las prácticas de estimación aplicables en las empresas participantes del estudio que hacen parte de las prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2, que los gerentes de proyecto consideran aplicables, existen diferentes niveles de aplicación.

Cu: definido en H2.

Cun: prácticas actuales de estimación en las empresas participantes del estudio que hacen parte de las prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2 y que los gerentes de proyecto consideran que tienen un nivel de aplicación parcial.

$$H3 : C-(Cu \cap Cun) \neq \emptyset$$

Hipótesis Alternativa (H4): Dentro de las prácticas de estimación planteadas en el modelo CMMI-DEV v1.2, y que no hacen parte de las practicas actuales de la empresa, existen prácticas que los gerentes de proyecto consideran conveniente implementar en la empresa.

D: prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2 que no hacen parte de las prácticas actuales de la empresa. $D = B-(A \cap B)$.

E: prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2 que no

hacen parte de las prácticas actuales de la empresa y que los gerentes de proyectos consideran conveniente implementar en la empresa.

$$H4 : D - (D \cap E) \neq \emptyset$$

Vista desde la adopción de métodos de estimación.

Hipótesis Nula (H0B): No existe diferencia en la precisión de la estimación entre empresas que usan métodos formales de estimación y las empresas que utilizan un método ad-hoc.

A: Precisión de la estimación de las empresas que utilizan métodos formales de estimación

B: Precisión de la estimación de las empresas que utilizan métodos ad-hoc de estimación

$$H0B : A = B$$

Hipótesis Alternativa (H1B): Existe diferencia en la precisión de la estimación entre las empresas que utilizan métodos formales de estimación y las empresas que utilizan un método ad-hoc.

A: Precisión de la estimación de las empresas que utilizan métodos formales de estimación

B: Precisión de la estimación de las empresas que utilizan métodos ad-hoc de estimación

$$H1B : A \neq B$$

3.6.2. Selección de variables

Variables independientes

- Prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2, ver Tabla 3.1 y Tabla 3.2.
- Métodos de estimación considerados más aplicables en la industria, ver Tabla 3.3.

Variables dependientes

- Porcentaje de las prácticas de estimación usadas actualmente en la empresa respecto al total de prácticas planteadas en el modelo de calidad CMMI-DEV v1.2 por parte de la empresa.

- El grado de aplicabilidad de las prácticas planteadas en el modelo de calidad CMMI-DEV v1.2, desde el punto de vista de los gerentes de proyectos.
- El nivel de adecuación de la especificación de las prácticas comunes entre la empresa y las prácticas de estimación planteadas en el modelo de calidad CMMI-DEV v1.2.
- El nivel de uso de los métodos formales y heurísticos de estimación por parte de las empresas.
- Precisión de la estimación del esfuerzo de las empresas participantes del estudio, de acuerdo a la ecuación planteada en [66].

EPR : Esfuerzo Promedio Real

EPE : Esfuerzo Promedio Estimado

$$\text{Precisión estimación del esfuerzo} = 1 - \left| \frac{(EPR - EPE)}{EPE} \right|$$

Precisión de la estimación de la duración de proyectos de software de las empresas participantes del estudio, de acuerdo a la ecuación planteada en [66].

DPR : Duración Promedio Real

DPE : Duración Promedio Estimada

$$\text{Precisión estimación de la duración} = 1 - \left| \frac{(DPR - DPE)}{DPE} \right|$$

3.6.3. Selección de personal

En el estudio experimental se involucraron a las empresas que hicieron parte del primer lote de Pymes participantes en el proyecto de mejora de procesos impulsado por la RCC-TIC [54] en Colombia durante los años 2008-2010.

Como representantes de cada empresa se seleccionara un gerente de proyecto con experiencia en las prácticas de estimación y una experiencia superior a 1 año en la empresa a la cual representa, con el objetivo de garantizar la autenticidad de los datos y disminuir el riesgo de baja motivación o inexperiencia en la realización de la encuesta.

Recursos Humanos

- Experimentadores (2)

- Instructores (2)

Recursos de Software

- Microsoft Office 2010 Professional
- Windows 7.0 Professional
- Base de datos en Excel
- Servicio de correo para envío de encuestas y recepción de encuestas diligenciadas

Recursos de Hardware

- 1 PC: Intel® Pentium Dual-Core™ E5500 w/ VT (2.8GHZ, 2MB L2, 800FSB), Windows® 7 Professional Original de 64 bit, Memoria de 4GB Dual Channel DDR3 SDRAM 1333MHz - 2DIMMs, Monitor Dell E1910H - Ancho y Plano de 18.5", Disco Duro de 500GB Serial ATA (7200RPM) DataBurst Caché™, Single Drive: 16X (DVD+/-RW) Burner Drive, 512MB NVIDIA® GeForce® G310 (DVI + VGA + HDMI).
- Recursos de Red: Local Area Network LAN Acceso a internet

3.7. Diseño del experimento

El estudio experimental se realizó a través de una encuesta que fue enviada a todas y cada una de las empresas de Bogotá, Medellín, Cali y Bucaramanga que participaron en el estudio experimental.

Las empresas fueron agrupadas según el número de empleados utilizando las mismas categorías utilizadas en el proyecto masivo de acompañamiento, [2]: micro-empresas (≤ 10), pequeñas micro (11–20), pequeña (21–50), mediana (51–99) y grandes empresas (≥ 100).

Dado que en encontramos datos extremos (out-liers) los cuales no cumplían con una distribución que pueda ser interpretada estadísticamente normal, se realizó un balanceamiento y eliminación de estos datos extremos para garantizar un mejor análisis. El estudio combina análisis tanto cualitativo como cuantitativo. El análisis sobre el proceso de estimación se llevó a cabo por medio de variables cualitativas, mientras que el análisis sobre la precisión de la estimación se realizó mediante variables cuantitativas.

3.7.1. Instrumentación

Los datos fueron recolectados a través de una encuesta que está estructurada en 5 secciones:

Sección 1. Información general de la organización.

Pregunta 1.1 ¿Cuántos años de constitución tiene la empresa?

- a. 2 años o menos
- b. De 3 a 5 años
- c. De 6 a 9 años
- d. 10 años o más

Pregunta 1.2. ¿Cuántos empleados tienen la empresa actualmente?

- a. 10 empleados o menos
- b. Entre 11 y 20 empleados
- c. Entre 21 y 50 empleados
- d. Entre 51 y 100 empleados
- e. Más de 101 empleados

Pregunta 1.3. ¿Qué estándares o modelos de calidad ha implementado la empresa?

- a. ISO-9000
- b. ISO/IEC SPICE
- c. CMMI-2
- d. CMMI-3
- e. CMMI-4
- f. CMMI-5
- g. Otro, ¿cuál?

Pregunta 1.4. ¿Cuántos años de experiencia tiene la empresa en la implementación o adopción de estándares o modelos de calidad orientados a la mejora de procesos?

- a. De 0 a 2 años
- b. De 3 a 4 años
- c. De 5 a 6 años
- d. 7 años o más

Sección 2. Información acerca de las prácticas de estimación.

Todas las preguntas de esta sección tienen como respuesta la siguiente escala:

- a. Práctica no aplicable en la empresa
- b. Práctica no utilizada en la empresa
- c. Práctica utilizada pocas veces
- d. Práctica utilizada en algunos proyectos
- e. Práctica institucionalizada en todos los proyectos.

Pregunta 2.1. ¿Se tienen definidos puntos de control para monitorear el progreso del proyecto con respecto a lo estimado?

Pregunta 2.2. ¿Cuál es el porcentaje de desviación del progreso del proyecto con respecto a lo estimado, a partir del cual se considera un desvío significativo y se procede a aplicar acciones correctivas? (El valor dado debe estar entre 0 % a 100 %)

Pregunta 2.3. ¿En caso de presentarse una desviación significativa, se realizan ajustes al cronograma y se estima nuevamente las tareas?

Pregunta 2.4. ¿Se descompone el alcance general del proyecto en unidades más pequeñas de trabajo (WBS)?

Pregunta 2.5. ¿La estimación del proyecto se realiza tomando como referencia la estructura de trabajo antes definida (WBS)?

Pregunta 2.6. ¿Durante el proceso de estimación de toman como referentes datos históricos de otros proyectos?

Pregunta 2.7. ¿La organización tiene definidas políticas claras con respecto a la estimación de los proyectos?

Pregunta 2.8. ¿En la planificación de los proyectos se incluye el tiempo requerido para realizar la estimación de los proyectos?

Pregunta 2.9. ¿La empresa cuenta con personas especializadas en la estimación de proyectos y con herramientas de software que apoyen el proceso de estimación?

Pregunta 2.10. ¿Se involucra en la estimación a las personas responsables de ejecutar las tareas?

Pregunta 2.11. ¿La empresa se preocupa por entrenar y capacitar al personal involucrado en la estimación con referente en nuevos métodos de estimación formales o basados en juicios de expertos?

Pregunta 2.12. ¿Al finalizar los proyectos se realiza un proceso de cierre donde se analiza la desviación de lo estimado versus lo ejecutado?

Pregunta 2.13. ¿Se realiza una revisión de la estimación por parte de una persona externa para evaluar si el método de estimación definido fue aplicado correctamente?

Pregunta 2.14. ¿La alta gerencia recibe información periódica del progreso de los proyectos comparando lo estimado versus lo ejecutado y su desviación?

Sección 3. Métodos de estimación utilizados

Pregunta 3.1. Para cada uno de los siguientes métodos de estimación(ver Tabla 3.4), seleccione el valor correspondiente según el nivel de conocimiento y/o aplicación en la empresa, de acuerdo a la siguiente convención:

- (1) No lo conozco
- (2) Lo conozco en teoría
- (3) Aplicado a un proyecto piloto
- (4) Aplicado a criterio del gerente de proyectos
- (5) Institucionalizado y calibrado de acuerdo al contexto de la organización

Pregunta 3.2. ¿Utiliza más de un método de estimación para el mismo proyecto, con el propósito de disminuir el nivel de incertidumbre de la estimación?

- a. Practica no aplicable en la empresa
- b. Practica no utilizada en la empresa
- c. Practica utilizada pocas veces
- d. Practica utilizada en algunos proyectos

Método de Estimación	Valor				
	1	2	3	4	5
A. Juicio de Expertos					
B. Delphi, Delphi Banda Ancha					
C. Puntos de Casos de Uso					
D. Puntos de Función Ajustados					
E. Basado en componentes genéricos					
F. Planning Poker					
G.COCOMO, COCOMO II					
H. Modelo Basado en Redes Neuronales					
I. Lógica Difusa (Fuzzy Logic)					
J. Otros modelos estadísticos					
K. Modelo propio de la empresa					

Tabla 3.4: Nivel de conocimiento de los métodos de estimación.

e. Practica institucionalizada en todos los proyectos.

Pregunta 3.3. ¿Utiliza distintos métodos de estimación para los diferentes tipos de proyecto?

- a. Practica no aplicable en la empresa
- b. Practica no utilizada en la empresa
- c. Practica utilizada pocas veces
- d. Practica utilizada en algunos proyectos
- e. Practica institucionalizada en todos los proyectos.

Sección 4. Precisión de las estimaciones de esfuerzo y duración.

Pregunta 4.1. ¿Cuántos proyectos fueron culminados en el año 2010?
(Pregunta abierta)

Pregunta 4.2. ¿Cuál fue el número promedio de integrantes en los proyectos culminados en el año 2010?
(Pregunta abierta)

Pregunta 4.3. ¿Cuál fue el tamaño promedio de los proyectos culminados en el año 2010? (Explique la unidad de medida que utiliza su organización)
(Pregunta abierta)

Pregunta 4.4. ¿Cuál fue el esfuerzo promedio estimado de los proyectos culminados en el año 2010? (En unidad de medida Horas/Hombre)

(Pregunta abierta)

Pregunta 4.5. ¿Cuál fue el esfuerzo promedio real de los proyectos culminados en el año 2010? (En unidad de medida Horas/Hombre)

(Pregunta abierta)

Pregunta 4.6. ¿Cuál fue la duración promedio estimada de los proyectos culminados en el año 2010? (En unidad de medida semanas)

(Pregunta abierta)

Pregunta 4.7. ¿Cuál fue la duración promedio real de los proyectos culminados en el año 2010? (En unidad de medida semanas)

(Pregunta abierta)

Sección 5. Percepciones generales

Pregunta 5.1. ¿Qué oportunidades de mejora tiene la compañía referente a la estimación de proyectos de software?

(Pregunta abierta)

Pregunta 5.2. ¿Qué factores de éxito considera usted tiene la compañía respecto a la actividad de estimación?

(Pregunta abierta)

Pregunta 5.3. ¿Cuál es su nivel de satisfacción con respecto a la manera en que se está llevando a cabo la estimación en la organización?

- a) Totalmente satisfecho
- b) Satisfecho
- c) Parcialmente satisfecho
- d) Insatisfecho
- e) Totalmente insatisfecho

3.7.2. Validez de resultados

Validez interna

Los resultados del estudio serán validos en el contexto de empresas de desarrollo de software o de base tecnológica en Colombia, que hayan iniciado algún proyecto de mejora de procesos

con respecto al estándar de calidad CMM-DEV v.1.2 y estén enfocadas a la mejora de la precisión de las estimaciones de esfuerzo y calendarización.

Validez externa

La generalidad de los resultados está enfocada al contexto Colombiano, ya que el estudio se realizó sobre empresas colombianas que participaron en un proyecto masivo de acompañamiento. Como trabajo futuro enmarcado en la RCC-TIC se pretende replicar el estudio en empresas colombianas que ya tengan alguna trayectoria en implementación del modelo; es también deseable que este estudio pueda ser reproducido en otros países de América Latina que tengan condiciones socio-económicas similares a Colombia (Perú, Chile, entre otros).

Validez de construcción

Se contó con personal capacitado para realizar el estudio experimental, tanto los experimentadores, asesores del estudio experimental y los participantes tienen formación académica en el área de ingeniería de software y gestión de proyectos. Se exigió que los participantes encargados de responder la encuesta tuvieran experiencia en la aplicación de prácticas de estimación y por lo menos 1 año de experiencia en la empresa a la cual representan para garantizar que estuvieran al tanto de los procesos institucionales de la organización.

Validez de conclusión

En la validez de conclusión la mayor amenaza está dada por el número de empresas que respondieron la encuesta, de igual manera la distribución de los datos pudo no tener una distribución normal, que es una condición requerida antes de aplicar estadística descriptiva. En este caso se depuraron y eliminaron datos extremos cuando fue necesario.

Capítulo 4

Análisis de los Datos Recolectados

4.1. Información general

Se realizó un contacto vía correo electrónico con los gerentes de las empresas de software que participaron en el proyecto, para solicitar su apoyo al estudio; se envió el cuestionario electrónico el cual debía ser diligenciado por un gerente de proyecto de la empresa que estuviese directamente involucrado en las actividades de estimación de software y contase con una experiencia mínima de 1 año dentro de la empresa, esto último con el fin de disminuir el riesgo que el representante no estuviera al tanto de las prácticas organizacionales sobre estimación.

Los datos fueron recolectados entre los meses de octubre y noviembre del año 2011, el cuestionario fue enviado a las 51 empresas que participaron en el proyecto, y se recibió respuesta de 13 empresas (26 %) sobre el total de la población seleccionada para este estudio; 6 de las empresas contestaron diciendo que no tenían disponibilidad para responder la encuesta y 5 correos finalmente no pudieron llegar a sus destinatarios por problemas de dirección de correo o porque la persona ya no estaba vinculada a la empresa.

La Figura 4.1, muestra la distribución geográfica de las encuestas recibidas respecto a la población total de empresas seleccionadas para este estudio; se puede observar que las encuestas recibidas forman una muestra equilibrada de la población.

Para calcular el nivel de confianza de los resultados, se utilizó el mismo criterio empleado en [66]. Para este efecto se considero la población como el número total de encuestas repartidas, 51 e-mails fueron enviados a igual numero de empresas, y el valor de la muestra representa el número de encuestas respondidas correctamente, las cuales fueron 13 en nuestro caso.

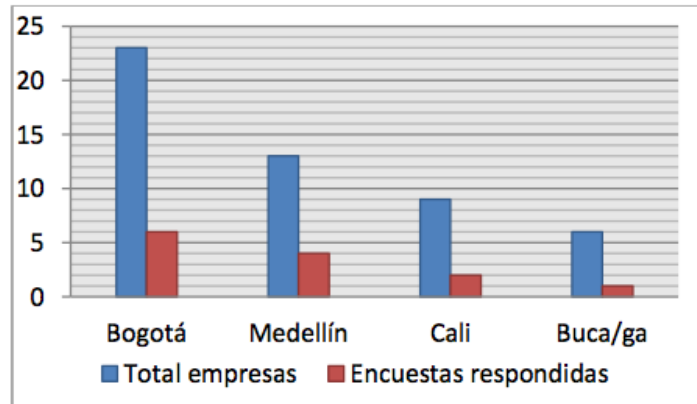


Figura 4.1: Distribución geográfica de las empresas que participaron en el estudio.

Se utiliza la siguiente formula, tomada de [26]:

$$E0 = \sqrt{\frac{N - n}{N * n}}$$

E0 = Nivel de confianza

N = Tamaño de la población

n = Tamaño de la muestra

Lo cual da como resultado un nivel de confianza del 76,1 %.

Las Figura 4.2 y Figura 4.3, muestran el perfil de las empresas que participaron en el estudio según el número de empleados y la experiencia en la adopción de modelos/estándares de mejora respectivamente.

4.2. Respuestas a las preguntas de investigación

Para realizar este análisis se tomaron las 14 prácticas de estimación extraídas del modelo de referencia CMMI-DEV v1.2, que están listadas en la Tabla 10, y se definió una pregunta en el cuestionario asociada directamente a cada práctica. Como ya fue mencionado las preguntas fueron formuladas de forma cualitativa y la respuesta podía tomar uno de los siguientes valores:

- a. Practica no aplicable en la empresa
- b. Practica no utilizada en la empresa

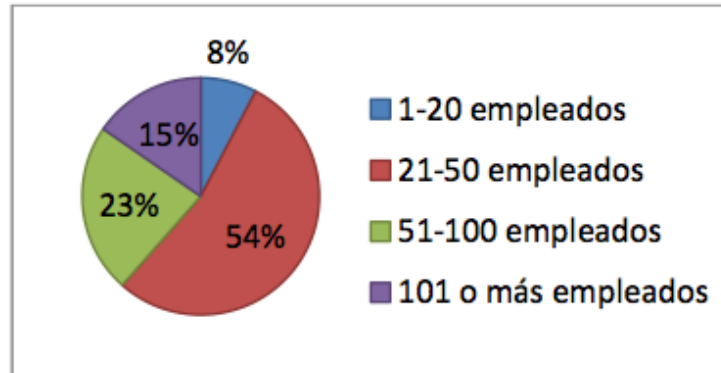


Figura 4.2: Distribución según número de empleados.

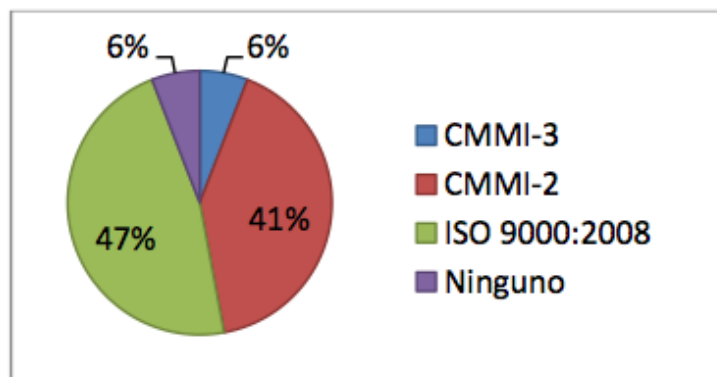


Figura 4.3: Distribución según Modelos/Estándares adoptados.

- c. Practica utilizada pocas veces
- d. Practica utilizada en algunos proyectos
- e. Practica institucionalizada en todos los proyectos.

Para efectos prácticos del análisis se agruparon las categorías (c) y (d) en una sola categoría denominada “Utilizada Ocasionalmente”. En la Tabla 4.1, se muestran las prácticas asociadas a la estimación las cuales surgieron de las practicas especificas y genéricas, tal como se explico en el capitulo anterior.

Id.	Práctica	Tipo de Práctica
P1	Monitorear los valores actuales de los parámetros de planeación del proyecto, tales como esfuerzo y duración del proyecto, comparándolos con los valores estimados en el plan de proyecto e identificar desviaciones significativas.	Específica
P2	En caso de existir una desviación significativa en la ejecución del proyecto de acuerdo a lo planeado se ejecutarán las acciones correctivas determinadas en el Plan del Proyecto.	Específica
P3	Establecer una estructura de ítems de trabajo a un nivel general para estimar el alcance del proyecto.	Específica
P4	Establecer y mantener los estimativos para las tareas y los ítems de trabajo.	Específica
P5	Estimar el esfuerzo de los productos de trabajo basados en una estimación a conciencia.	Específica
P6	Establecer y mantener una política organizacional para la planeación y el desempeño del proceso de estimación de proyectos de software.	Genérica
P7	Establecer y mantener un tiempo requerido para realizar la estimación en la planificación de los proyectos.	Genérica
P8	Proveer los recursos adecuados para desarrollar actividades de estimación planeadas, tales como: estimadores expertos, datos históricos, productos de trabajo, listas de tareas, etc.	Genérica
P9	Asignar autoridad y responsabilidad a los estimadores e involucrar a todos los participantes del proyecto.	Genérica
P10	Entrenar a los participantes del proyecto en la metodología de estimación, modelos, datos y/o algoritmos usados para determinar la estimación.	Genérica
P11	Identificar e involucrar en la estimación a las personas responsables de ejecutar las tareas.	Genérica
P12	Monitorear y controlar el progreso real del proyecto contra lo estimado y establecer criterios de acción correctiva en caso de una desviación importante del plan del proyecto.	Genérica
P13	Evaluar objetivamente la adherencia del proceso de estimación usando modelos paramétricos de estimación o modelos de juicio de expertos basados en datos históricos y controlar posibles adaptaciones del proceso.	Genérica
P14	Revisar el estado de las actividades y los resultados del proceso de estimación con la alta gerencia y determinar cuál es su nivel de satisfacción con respecto a la manera en que se está llevando a cabo la estimación en la organización.	Genérica

Tabla 4.1: Distribución de las empresas participantes en el proyecto.

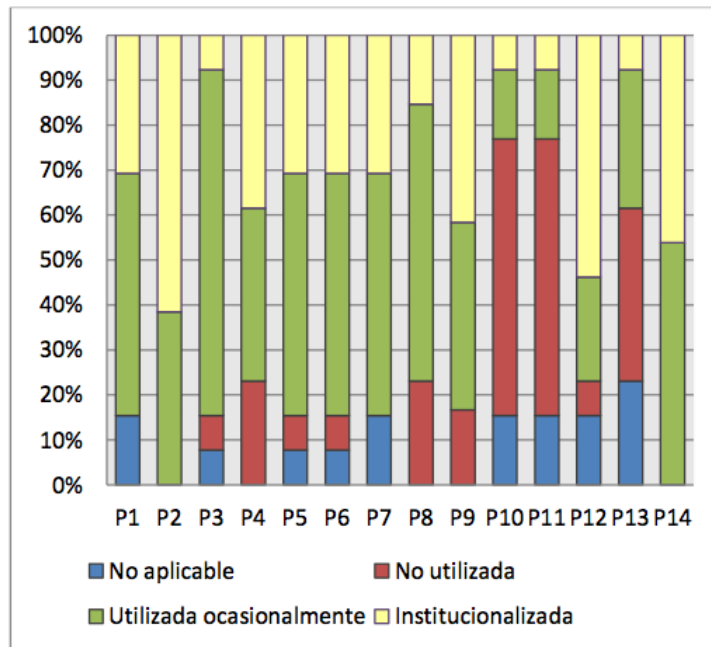


Figura 4.4: Frecuencia encontrada de las prácticas de estimación CMMI-DEV v1.2.

La Figura 4.4, muestra las 14 prácticas recomendadas por el modelo CMMI-DEV v1.2 y su frecuencia de uso encontrada en las empresas. A partir de esta información se pasa a resolver las preguntas correspondientes:

4.2.1. RQ1. ¿Cuáles son las prácticas de estimación más aplicadas en las empresas?

Según los resultados presentados en la Figura 4.4, se encuentra que las tres prácticas con mayor frecuencia de uso son las siguientes:

- **P2** – En caso de existir una desviación significativa en la ejecución del proyecto de acuerdo a lo planeado se ejecutaran las acciones correctivas determinadas en el Plan del Proyecto. Con una frecuencia de uso institucionalizada, en el 62 % de las empresas.
- **P12** – Monitorear y controlar el progreso real del proyecto contra lo estimado y establecer criterios de acción correctiva en caso de una desviación importante del plan del proyecto. Con una frecuencia de uso institucionalizada, en el 38 % de las empresas.

- **P14** – Revisar el estado de las actividades y los resultados del proceso de estimación con la alta gerencia y determinar cuál es su nivel de satisfacción con respecto a la manera en que se está llevando a cabo la estimación en la organización. Con una frecuencia de uso institucionalizada, en el 46 % de las empresas.

Según los resultados del estudio se encuentra que las tres prácticas con menor frecuencia de uso son las siguientes:

- **P10** – Entrenar a los participantes del proyecto en la metodología de estimación, modelos, datos y/o algoritmos usados para determinar la estimación. Esta práctica no está siendo utilizada, en el 62 % de las empresas.
- **P11** – Identificar e involucrar en la estimación a las personas responsables de ejecutar las tareas. Esta práctica no está siendo utilizada, en el 62 % de las empresas.
- **P13** – Evaluar objetivamente la adherencia del proceso de estimación usando modelos paramétricos de estimación o modelos de juicio de expertos basados en datos históricos y controlar posibles adaptaciones del proceso. Esta práctica no está siendo utilizada, en el 38 % de las empresas. Asimismo esta es la práctica que obtuvo mayor número de respuestas en la categoría No aplicable.

Análisis Adicional:

Con respecto a las prácticas menos utilizadas vale la pena notar que las prácticas P10 y P11 están relacionadas directamente con el recurso humano involucrado en la estimación, mostrando dos falencias básicas: no se da entrenamiento a las personas en la manera de estimar y no se involucra a los responsables de la tarea en la estimación.

Si se considera las categorías de prácticas específicas y practicas genéricas bajo la perspectiva de CMMI-DEV v1.2, se esperaría que las prácticas específicas tuviesen en general un nivel de aplicación mayor que las genéricas. Llama especialmente la atención el comportamiento de la práctica P3 (Establecer una estructura de ítems de trabajo a un nivel general para estimar el alcance del proyecto) que tuvo el nivel de institucionalización más bajo.

Esta justamente es la práctica de CMMI que recomienda de manera directa la necesidad de que la organización cuente con unidades de medida de tamaño del producto software que sirva como base para la estimación.

4.2.2. RQ2. ¿Cuáles es el nivel de aplicación de las prácticas consideradas útiles para la organización?

Para realizar el análisis del nivel de aplicación de las prácticas consideradas útiles dentro de la organización, el primer paso fue descartar aquellas practicas que en la mayoría (51 %

o más) de empresas fueron consideradas de categorías (a) No aplicables y (b) No utilizadas dentro de la organización. Por esta razón se descartaron para este análisis las prácticas P10, P11 y P13. Que consistentemente con el punto anterior fueron aquellas con menor frecuencia de uso.

Las prácticas restantes son consideradas útiles dentro de este estudio. Entiéndase útiles aquellas practicas que en la mayoría de las empresas (51 % o más) fueron catalogadas en las categorías (c), (d) agrupadas en Utilizadas ocasionalmente o en la categoría (e) Institucionalizadas.

Las prácticas que se encuentran institucionalizadas en la mayoría de las empresas (51 % o más) son las siguientes:

- **P2** – En caso de existir una desviación significativa en la ejecución del proyecto de acuerdo a lo planeado se ejecutarán las acciones correctivas determinadas en el Plan del Proyecto. Con una frecuencia de uso institucionalizada, en el 62 % de las empresas.
- **P12** – Monitorear y controlar el progreso real del proyecto contra lo estimado y establecer criterios de acción correctiva en caso de una desviación importante del plan del proyecto. Con una frecuencia de uso institucionalizada, en el 54 % de las empresas.

El conjunto de prácticas restantes, el cual está conformado por las prácticas: P1, P3, P4, P5, P6, P7, P8, P9 y P14. Se puede concluir que son utilizadas ocasionalmente de acuerdo al criterio del gerente del proyecto o a las necesidades del entorno, pero no alcanzan a estar institucionalizadas en la mayoría de las empresas que participaron del estudio.

4.2.3. RQ3. ¿Cuál es la precisión de estimación del esfuerzo en los proyectos de software?

Para realizar el análisis de la precisión del esfuerzo se utilizó la misma perspectiva del estudio iMPS [66]. De modo que se les pidió a las empresas que tomaran los datos históricos de los proyectos culminados durante el año 2010, y con base a estos datos que respondieran las siguientes preguntas:

- a. ¿Cuál fue el esfuerzo promedio estimado de los proyectos culminados en el año 2010, en Horas/Hombre?
- b. ¿Cuál fue el esfuerzo promedio real de los proyectos culminados en el año 2010, en Horas/Hombre?

De donde se obtuvieron los valores de las siguientes variables: EPR: Esfuerzo Promedio Real, EPE: Esfuerzo Promedio Estimado. Con estos dos valores se utiliza la siguiente

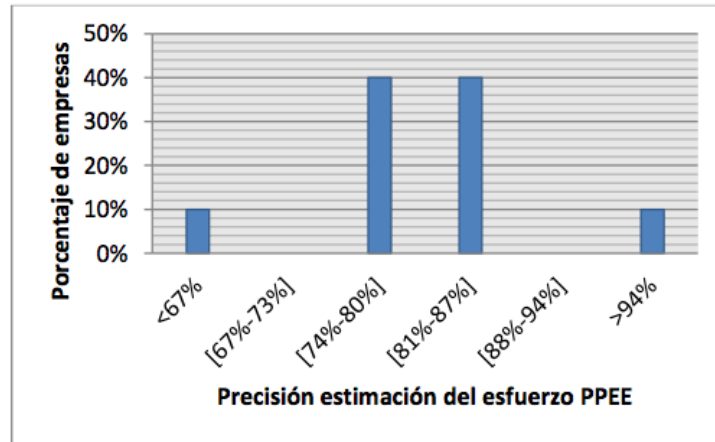


Figura 4.5: Distribución del porcentaje de precisión de la estimación de esfuerzo (PPEE)

fórmula para obtener el Porcentaje de Precisión de la Estimación del Esfuerzo, PPEE, de acuerdo a [66]:

$$\text{Porcentaje de Precisión Estimación del Esfuerzo (PPEE)} = 1 - \left| \frac{(EPR - EPE)}{EPE} \right|$$

En la Figura 4.5, se pueden identificar los valores calculados para la precisión de la estimación del esfuerzo en las empresas encuestadas; a pesar de contar con un conjunto pequeño de datos, se observó una tendencia normal en la distribución con un valor de la media del 80 % y una desviación estándar del 7 %, de tal forma que el 80 % de los datos se sitúan en el rango central y el 20 % restante en los extremos.

El nivel de confianza de este análisis se vio afectado debido a que 3 de las 13 encuestas recibidas, no brindaron información referente a la precisión de la estimación del esfuerzo. De lo cual se obtuvo un nivel de confianza del 71.7 %.

4.2.4. RQ4. ¿Cuál es la precisión de estimación de la calendarización en los proyectos de software?

El análisis de la precisión de la duración o calendarización de proyectos se realizó de manera similar al anterior, también se les pidió a las empresas que tomaran como referencia los datos históricos de los proyectos culminados en el año 2010, y con base a estos datos que respondieran las siguientes preguntas:

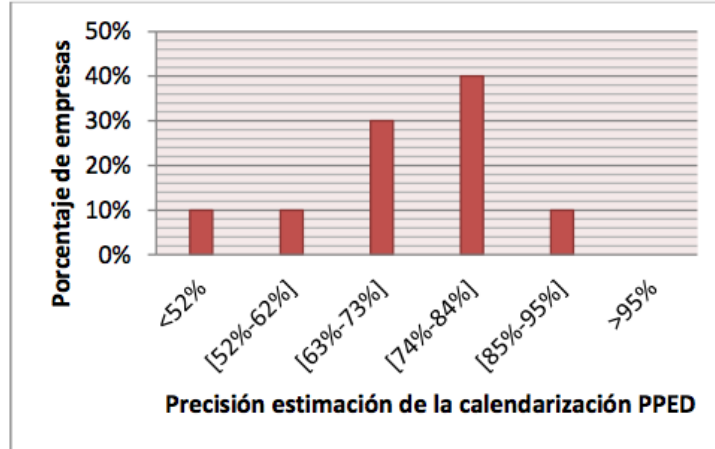


Figura 4.6: Distribución del porcentaje de precisión de la estimación de la duración (PPED)

- ¿Cuál fue la duración promedio estimada de los proyectos culminados en el año 2010, en unidades de semanas?
- ¿Cuál fue la duración promedio real de los proyectos culminados en el año 2010, en unidades de semanas?

De donde se obtuvieron los valores de las siguientes variables: DPR: Duración Promedio Real, DPE: Duración Promedio Estimada. Con estos dos valores se calcula la siguiente fórmula para obtener el Porcentaje de Precisión de la Estimación de la Duración, PPED, de acuerdo a [66]:

$$\text{Porcentaje de precisión estimación de la duración (PPED)} = 1 - \left| \frac{(DPR - DPE)}{DPE} \right|$$

En la Figura 4.6, se identifican los valores calculados para la precisión de la estimación de la duración de proyectos en las empresas encuestadas; asumiendo una distribución normal se encontró que la media está ubicada en 73 % y una desviación estándar del 11 %, de tal forma que el 70 % quedo ubicada en el rango central, a más y menos una desviación estándar, 20 % quedo en dentro del rango entre la media más y menos dos desviaciones estándar y tan solo el 10 % quedó ubicado en el extremo inferior negativo.

El nivel de confianza de este análisis se vió afectado debido a que 3 de las 13 encuestas recibidas, no brindaron información referente a la precisión de la estimación de la duración. De lo cual se obtuvo un nivel de confianza del 71.7 %.

4.2.5. RQ5. ¿Cuáles son las oportunidades de mejora y factores de éxito que perciben las compañías en el proceso de estimación?

Esta sección del cuestionario recopiló las respuestas a las preguntas abiertas realizadas al personal de gerencia de proyectos encargados de la estimación, y se les pidió que dieran su percepción acerca de las oportunidades de mejora que tiene la compañía, y los factores de éxito que consideran han tenido hasta este momento respecto a la actividad de estimación.

Como factores de éxito, encontramos los siguientes:

- Tener personal con experiencia en gerencia de proyectos, que a través de Juicio de experto, generaron el modelo que actualmente usa la organización el cual diferencia bien las distintas funcionalidades a entregar propias de la empresa.
- Estandarizar los proyectos a un modelo de Servicios y Procesos, los cuales permitan hacer una estimación más precisa de acuerdo a proyectos anteriores de condiciones similares.
- Al tener una estimación adecuada y adaptada a los procesos propios de la organización y conociendo lo que realmente se va a utilizar en el proyecto, se genera rentabilidad.
- La experiencia de algunos ingenieros en las funcionalidades trabajadas, lo que brinda alta productividad y agilidad en la ejecución de las tareas.
- Migrar de estimación de juicio experto, a la construcción de modelos a priori acordados basados en la unidad de tamaño del servicio.

Como oportunidades de mejora, encontramos las siguientes:

- La creación de un repositorio histórico de duración, esfuerzo y costos de los proyectos al momento de la estimación, tanto como en su fase de terminación; se pueden obtener modelos estadísticos para ajustar el modelo que actualmente usa la organización. Es una tarea que va en paralelo con la ejecución de los proyectos en la empresa.
- Ajustar los procesos del servicio de Administración de Proyectos. Esto implica la utilización de metodologías de PMI y CMMI.
- No se pueden mezclar la precisión de la estimación en el calendario, con la precisión de la estimación del esfuerzo para determinar si el proyecto fue concluido exitosamente. Si un proyecto termina dentro del tiempo acordado pero a costa de un sobre esfuerzo del equipo de trabajo, no debería ser considerado exitoso, el verdadero éxito

de la estimación es tratar de predecir ambas dimensiones y que estén ajustadas a la realidad.

- Lo primero es estructurar diferentes tipos de proyecto ya que actualmente no se tiene este esquema. Lo segundo se requiere profundizar en las fases de levantamiento de requerimientos de negocio para lograr una WBS más detallada y obtener claridad en las tareas a realizar para cada work package. Lo tercero se debe entrenar a los directores de proyecto en técnicas de estimación. Cuarto se debe iniciar un levantamiento estadístico para obtener en proyectos futuros una estimación con desviaciones menos significativas. Quinto se debe ser más estricto en los seguimientos y controles de cambio que afectan el alcance/tiempo/recursos/riesgos/QA.
- Implementar procesos de aseguramiento de la calidad en la gestión de proyectos para garantizar el seguimiento y evaluación de resultados.

4.2.6. RQ6. ¿Cuál es el nivel de aplicación en la organización de los métodos de estimación reconocidos en la literatura?

Para realizar este análisis se seleccionaron 10 métodos de estimación, con base en [45], los cuales se muestran en la Tabla 9. Las preguntas acerca de la frecuencia del uso de los métodos de estimación fueron formuladas de forma cualitativa y la respuesta podía tomar uno de los siguientes valores:

- 1 = No lo conozco
- 2 = Lo conozco en teoría
- 3 = Aplicado a un proyecto piloto
- 4 = Aplicado a criterio del gerente de proyectos
- 5 = Institucionalizado y calibrado de acuerdo al contexto de la organización

Igualmente, para propósitos prácticos, las opciones (3) y (4) se agruparon en una sola categoría denominada “Aplicado Ocasionalmente”.

La Figura 4.7, muestra los 10 métodos de estimación más comunes [45] y su aplicación de uso dentro de las empresas.

De la figura anterior, se resalta que los métodos más usados, combinando las categorías institucionalizado y aplicado ocasionalmente, en las empresas son: el Juicio de Expertos, puesto en práctica en el 92 % de las empresas; se puede notar también que el 54 % de las empresas cuentan con un método propio de estimación y los métodos de estimación: Puntos de Función Ajustados 54 % y Puntos de Casos de Uso 46 %, son considerablemente utilizados por las Pymes de desarrollo de software.

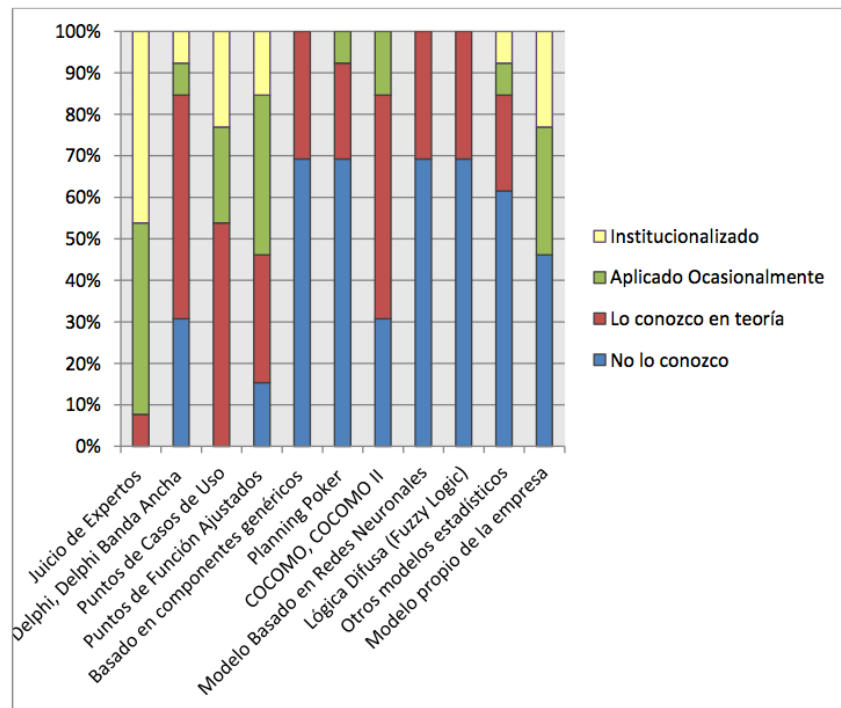


Figura 4.7: Aplicación de los métodos de estimación

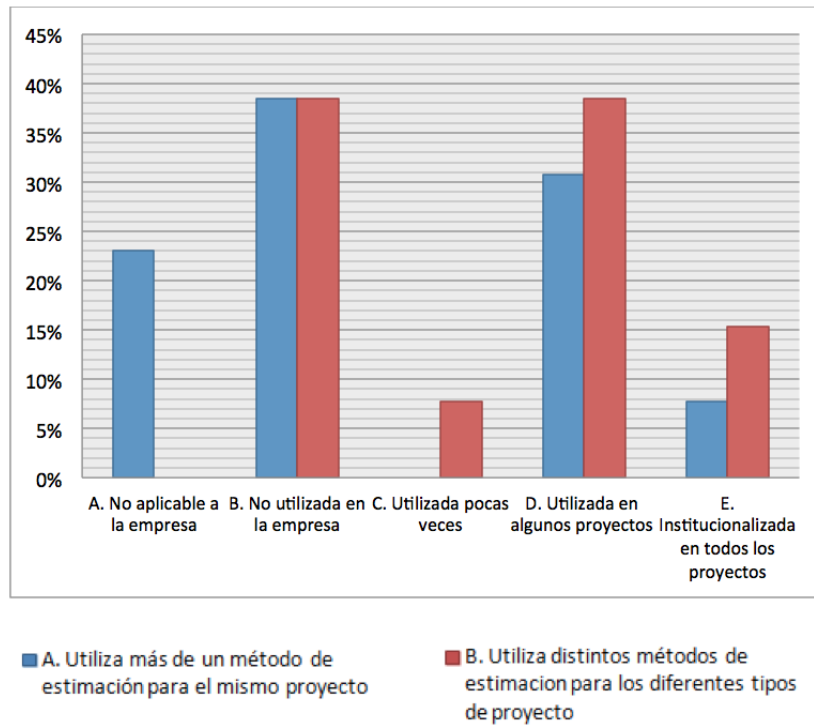


Figura 4.8: Nivel de aplicación de los métodos de estimación

Otro aspecto a resaltar es, evidenciar la poca acogida que tienen en nuestras empresas métodos basados en parámetros matemáticos, que utilizan regresiones lineales o lógica difusa, métodos tales como: fuzzy logic, modelos basados en redes neuronales o modelos basados en componentes genéricos alcanzaron una calificación pobre respecto a su difusión y uso en las Pymes colombianas.

4.2.7. RQ7. ¿De qué manera se aplican en la organización los métodos de estimación?

La Figura 4.8, responde a las siguientes preguntas puntuales:

A. ¿Utiliza más de un método de estimación para el mismo proyecto, con el propósito de disminuir el nivel de incertidumbre de la estimación?

B. ¿Utiliza distintos métodos de estimación para los diferentes tipos de proyecto?

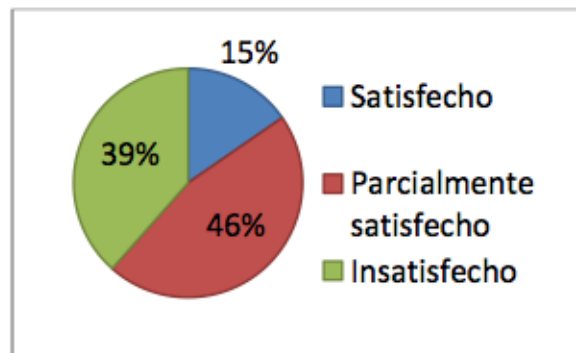


Figura 4.9: Nivel de satisfacción con el proceso de estimación.

De la anterior figura se observa menos del 10 % de las empresas involucradas en el estudio utiliza mas de un método de estimación para disminuir el porcentaje de error de la estimación y tan solo cerca del 15 % utiliza métodos diferentes adaptados a cada proyecto. En conclusión se observa que todavía falta mucha conciencia por parte de las Pymes acerca de la importancia de adoptar métodos de estimación en sus prácticas organizacionales.

4.2.8. RQ8. ¿Cuál es el nivel de satisfacción de los representantes de la organización con respecto a la estimación?

Para analizar el nivel de satisfacción con respecto a la manera en que se está llevando a cabo la estimación en la organización, los participantes escogieron una, entre las siguientes opciones:

- a. Totalmente satisfecho
- b. Satisfecho,
- c. Parcialmente satisfecho
- d. Insatisfecho
- e. Totalmente Insatisfecho

La Figura 4.9, muestra los resultados encontrados en esta pregunta. Donde se resalta que ninguna de las empresas expresa encontrarse totalmente satisfecho, el 15 % de las empresas encuestadas expresaron sentirse satisfechas con el proceso de estimación, un 46 % de las empresas se encuentra parcialmente satisfecho, mientras que el restante 39 % manifiesta sentirse insatisfecho. Por esta razón se evidencia que existen grandes oportunidades de mejora en los procesos de estimación.

Análisis Adicional:

Se puede notar que las empresas colombianas que se ubican en el rango de la media más ó menos una desviación del esfuerzo, son aquellas que tienen institucionalizadas ó aplican frecuentemente entre 7 a 10 de las 14 prácticas recomendadas por el modelo CMMI-DEV v1.2; Asimismo, se observó que las empresas colombianas que tienen una alta precisión en la estimación del esfuerzo (más del 94 %) utilizan por lo menos 3 métodos de estimación calibrados a su contexto organizacional.

4.3. Análisis de hipótesis

Debido al tamaño de muestra pequeño, no fue posible aplicar un análisis estadístico riguroso, tal como una prueba de hipótesis para probar las hipótesis cuantitativas planteadas en la sección 3.3.1. Sin embargo, se aplicó estadística descriptiva para interpretar los datos recolectados y darles forma, con la intención de entender la población a la cual representa la muestra.

4.4. Limitaciones

Una de las principales limitaciones del estudio, es el número de encuestas recibidas. Si bien este porcentaje de encuestas puede considerarse una muestra esperada, de acuerdo con otras encuestas normalmente realizadas, es muy deseable que un mayor número de empresas pueda apoyar este estudio para aumentar el nivel de confianza para alcanzar, por lo menos, un 85 % de acuerdo con [65].

Para realizar el análisis cuantitativo de la precisión de la estimación del esfuerzo y la duración de proyectos de software, se eliminaron datos extremos que afectaban el comportamiento normal de la distribución. Los escenarios que representaron comportamientos extremos fueron los siguientes:

- (a) reporte de una precisión del 100 % en las estimaciones tanto de esfuerzo como de calendario, lo que se considera un escenario alejado de la realidad de las empresas.
- (b) Desviación del esfuerzo muy alta 1500 % (80 h.h. / 1280 h.h).
- (c) Sin reporte de información en las preguntas de estimación

Queda por fuera del alcance del estudio garantizar la veracidad de los datos cuantitativos de estimación reportados por los representantes de las empresas; en este tipo de información cuantitativa sensible recopilada a través de una encuesta virtual, se corre este riesgo.

Capítulo 5

Conclusiones y Trabajos Futuros

5.1. Conclusiones

Se han presentado los resultados de un estudio acerca del estado de la estimación en empresas de desarrollo de software colombianas que han adoptado recientemente el modelo CMMI-DEV v1.2.

A partir de los resultados obtenidos, y a pesar del bajo número de encuestas procesadas se lograron identificar tendencias y oportunidades de mejora relevantes, complementarias a las planteadas en la sección 4.2.5, las cuales se describen a continuación:

- El bajo nivel de institucionalización de (P13 – Evaluar objetivamente la adherencia del proceso de estimación usando modelos paramétricos de estimación o modelos de juicio de expertos basados en datos históricos y controlar posibles adaptaciones del proceso), muestra que la mayor parte de las organizaciones con cuenta con un proceso formal establecido para realizar la estimación. Uno de los aspectos claves para iniciar la formalización de este proceso es la disciplina que la organización debe generar ya que esta práctica no está siendo utilizada, en el 38 % de las empresas. Es importante enfocar los esfuerzos a fortalecer la base de datos histórica para integrar modelos estadísticos de predicción que permitan una mayor precisión en las estimaciones.
- El uso intensivo del método de juicio de experto y el bajo uso de métodos de tipo paramétrico, demuestra que la organización no ha desarrollado confianza en un modelo de estimación pues es altamente dependiente de las personas. Esta alta dependencia de los expertos para estimar conlleva un alto riesgo de crisis cuando no se pueda contar con el (los) experto(s). La experiencia de estimación es uno de los conocimientos más valiosos de la organización que, como ya fue mencionado, representa una ventaja competitiva; es por eso recomendable que gradualmente la organización vaya depen-

diendo menos de una persona (conocimiento tácito solo en la mente de los expertos) y vaya estableciendo un modelo que representa la realidad de la organización, es decir, hacer explícito el conocimiento de los expertos.

- Es importante enfocar los esfuerzos a fortalecer la base de datos histórica para integrar modelos estadísticos de predicción que permitan una mayor precisión en las estimaciones. En caso de no tener una base histórica propia, se sugiere utilizar aquellas bases de conocimiento internacionales que son accesibles, filtrando los proyectos por aquellos que se asemejen a las condiciones particulares de la empresa, esto servirá como un punto de partida.
- Otras de las debilidades detectadas en el estudio se refiere a que, en general, no se planifica el tiempo previsto para la estimación, no se capacita a los directores en técnicas de estimación y se da baja participación en la estimación a las personas responsables de las tareas. Estas falencias están directamente relacionadas con una baja institucionalización del proceso de estimación y las organizaciones deben trabajar en ese sentido si desean avanzar en la madurez de sus procesos.
- Un aspecto positivo es el hecho de que las empresas reconocen que aun tienen muchos aspectos por mejorar en cuanto a la estimación. De otra parte, se reconoce una responsabilidad importante de la academia para generar espacios en los que se divulguen y experimenten los diferentes modelos de estimación.
- Se destaca también el papel de los referentes de calidad para recomendar prácticas de apoyo a la estimación. En el análisis general realizado se encontró que las Pymes con mejor desempeño en la estimación de sus proyectos son aquellas que presentan madurez más alta y hacen una combinación de prácticas y métodos de estimación. Si bien las prácticas referentes para este estudio fueron obtenidas de CMMI-DEV, una buena aplicación de los principios de estimación está fundamentada en principios y estudios de la ingeniería de software que van más allá de un modelo de calidad u otro.
- Los resultados de este estudio están ceñidos al contexto de empresas de desarrollo de software colombianas que recientemente hicieron un primer ciclo de mejora utilizando CMMI como modelo referente. Es posible que los resultados puedan ser muy diferentes en empresas que ya han tenido más de un ciclo de mejora con este proceso. Sin embargo, una investigación adicional necesita ser realizada en Colombia.

5.2. Trabajos futuros

Este estudio empírico realizado en el tema de la estimación, es uno de los primeros estudios en Colombia para la industria de software que se realiza con la formalidad esperada de un estudio experimental. Sin embargo, este trabajo de maestría es solo la plataforma de inicio

para continuar en este trabajo de impactar la industria de software de tal manera que sea más competitiva. Algunos de los trabajos que dan continuidad a este estudio, son los siguientes:

- Realizar un estudio focalizado al conjunto de empresas referentes que presentaron mejores resultados, es decir, que se encuentran en el rango de desviación menor, con el propósito de aprender de estas empresas. Una motivación especial en este estudio es analizar el (los) método (s) de estimación utilizados con el propósito de contar con evidencia objetiva de métodos de estimación aplicados exitosamente que se ajustan a las condiciones particulares de las empresas colombianas.
- Realizar una replicación de este estudio en otras empresas en Colombia que ya cuentan con alguna trayectoria en ciclos de mejora utilizando CMMI como referentes. Los estudios sucesivos debe ser contrastado con los resultados de este estudio y en lo posible, agregados con el propósito de definir el perfil de estimación de las empresas en Colombia.
- Asimismo es deseable realizar este estudio en otros países de América Latina que tengan condiciones similares a Colombia y estén dispuestos a contribuir, tales como México, Perú y Chile.

5.3. Publicaciones relacionadas.

- Páez Anaya, Iván Darío, Anaya Raquel. Travassos H. Guilherme. Estado Actual de la Estimación de Software en Compañías Colombianas que han Adoptado CMMI. Experimental Software Engineering Latin American Workshop. ESELAW 2012. [49].

Bibliografía

- [1] *American Heritage Dictionary*. Second College Edition, 1985.
- [2] Anaya, Raquel. Gómez, Liliana. *Lecciones Aprendidas en el Acompañamiento Masivo para Mejora de Procesos en Empresas de Software: Un Caso Colombiano*. Memorias XV Congreso Iberoamericano de Ingeniería de Software CIBSE. Buenos Aires, Argentina. Abril, 2012.
- [3] Anda Bente, Hege Dreiem, Dag I.K. Sjøberg, Magne Jørgensen. *Estimating Software Development Effort based on Use Cases - Experiences from Industry*. 4th International Conference on the Unified Modeling Language UML2001, Springer- Verlag. Toronto, Canada. October 1-5, 2001.
- [4] Anda Bente. *Comparing Effort Estimates Based on Use Case Points with Expert Estimates*. Empirical Assessment in Software Engineering (EASE 2002). Keele, UK. April 8-10, 2002.
- [5] Araujo, M. A. ; Barros, M. O.; Murta, L. G. P.; Travassos, G.H. *Statistical Methods Applied to Experimental Software Engineering*. Tutorial at Brazilian Symposium on Software Engineering 2006. ESELaw 2009 Brazil. 2009.
- [6] Basili, V. R. *The experimental paradigm in software engineering*. In *Experimental Software Engineering Issues: Critical Assessment and Future Directives*. Proc of Dagstuhl-Workshop, H. Dieter Rombach, Victor R. Basili, and Richard Selby (eds), published as Lecture. Notes in Computer Science p706, Springer-Verlag 1993.
- [7] Basili, V. R. *The Past, Present, and Future of Experimental Software Engineering*. ACM-IEEE International Symposium on. Empirical Software Engineering - ISESE 2007.
- [8] Bergeron, F. and St-Arnaud, J.-Y. *Estimation of Information Systems Development Efforts: A Pilot Study*. Information & Management. V.22, p239-254. 1992.

- [9] Boehm B, Chris Abts, A. Winsor Brown, Sunita Chulani, Bradford K. Clark, Ellis Horowitz, Ray Madachy, Donald J. Reifer, and Bert Steece. *Software Cost Estimation with COCOMO II*. Englewood Cliffs, NJ:Prentice-Hall, 2000.
- [10] Boehm B., Abts C., Chulani, S. *Software development cost estimation approaches A survey*. Ann. Softw. Eng., 10(1-4), pp. 177-205. 2000.
- [11] Capretz, Luiz Fernando. Marza, Venus. *Improving effort estimation by voting software estimation models..* Advances in Software Engineering archive. Volume 2009, January 2009.
- [12] Carmen G., Maria. Garzás, Javier . *Método de Estimación de Puntos de Caso de Uso*. Kybele Consulting. <http://www.kybeleconsulting.com/recursos/articulos/estimacion-puntos-caso-de-uso/> Visitado en 2012.
- [13] CMMI for Development. *CMU/SEI-2010-TR-033*. Software Engineering Institute, Carnegie Mellon University. 2010.
- [14] COCOMO II. *Center for Systems and Software Engineering*. http://sunset.usc.edu/csse/research/COCOMOII/cocomo_main.html/ Visitado en 2012.
- [15] Condori-Fernández, Nelly. *Un Procedimiento de Medición de Tamaño Funcional para Especificaciones de Requisitos*. Tesis de Doctorado. Universidad Politécnica de Valencia, UPV. España. Diciembre de 2006.
- [16] COSMIC. *Common Software Measurement International Consortium. Organization*. <http://www.cosmicon.com/> (Visitado en 2012).
- [17] COSMIC-FFP. *ISO/IEC 19761:2003 Software engineering. A functional size measurement method..* 2003.
- [18] Crespo F. *Un Modelo Paramétrico Matemático Difuso para la Estimación del Esfuerzo de Desarrollo del Software*. Tesis de doctorado. Escuela Politécnica Universidad de Alcalá, 2003. Disponible en <http://www.fdi.ucm.es/profesor/jcrespo/ante.pdf>. Fecha de consulta: Mayo 2012.
- [19] Da Yang, Qing Wang, Mingshu Li, Ye Yang, Kai Ye and Jing Du. *A Survey on Software Cost Estimation in the Chinese Software Industry*. Published by ACM 2008 Article, 2008.
- [20] Dybå, T., Kitchenham, B.A., and Jørgensen, M. *Evidence-Based Software Engineering for Practitioners*. IEEE Software, 22(1):58–65. 2005.
- [21] EDT. *Estructura de Descomposición del Trabajo*. http://es.wikipedia.org/wiki/Estructura_de_descomposicion_del_trabajo Visitado en 2012.

- [22] FiSMA. *Finnish Software Measurement Association*. <http://www.fisma.fi/> (Visitado en 2012).
- [23] FiSMA FSM. *FiSMA 1.1 functional size measurement method*. ISO/IEC 29881:2008 Information technology - Software and systems engineering. 2008.
- [24] FiSMA. *Functional Size Measurement Method version 1.1. Method Specification Document*. FSM Working Group. 2004.
- [25] Fuzzy Logic. *Softnet*. <http://www.soft-net.at/softnetwiki/bin/view/Main/Fuzzy+Logic> (Visitado en 2012).
- [26] Gardner, M.J; Altman, D. G. *Statistics with Confidence: confidence intervals and statistical guidelines*. London: BMJ Publishing Group. 1989.
- [27] Garmus, David and Herron, David *Function Point Analysis: Measurement Practices for Successful Software Projects*. Ed. Addison-Wesley; December 2000.
- [28] Heemstra, F.J. *Software cost estimation*. Information and Software Technology. 34, 10, 627-639. 1992.
- [29] Hipótesis nula. http://es.wikipedia.org/wiki/Hipotesis_nula (visitado 2012)
- [30] IEEE. *Standard for Application and Management of the Systems Engineering Process*. IEEE Std 1220-2005 (Revision of IEEE Std 1220-1998)
- [31] IFPUG. *International Function Point Users Group*. <http://www.ifpug.org/> (Visitado en 2012).
- [32] IFPUG FSM Method. *IFPUG functional size measurement method..* ISO/IEC 20926:2009 Software and systems engineering - Software measurement. Functional Sizing Methods. 2009
- [33] ISBSG. *International Software Benchmarking Standards Group*. <http://www.isbsg.org> (Visitado en 2012).
- [34] ISO/IEC 14143-1:2007. *Information technology. Software measurement. Functional size measurement*. Part 1: Definition of concepts. 2007
- [35] Jenkins, A.M., Naumann, J.D. and Wetherbe, J.C. *Empirical Investigation of Systems Development Practices and Results*. Information and Management, 7: p. 73-82. 1984.
- [36] Jorgensen, M. and Molokken, K. *How large are software cost overruns? A review of the 1994 Chaos Report..* Information and Software Technology. 48, 4. April. 2006.
- [37] Juristo N. Moreno A, et al. *Basics of Software Engineering Experimentation*. Kluwer Academic Publishers, 2000.

- [38] Kitchenham, B.A. Pfleeger, S.L., Pickard, L.M., Jones, P.W., Hoaglin, D.C., El Emam, K., and Rosenberg, J., *Preliminary Guidelines for Empirical Research in Software Engineering*. IEEE Trans. Software Eng., vol. 28, no. 8, pp. 721–734. 2002.
- [39] Lederer, A.L. and Prasad, J. *Causes of Inaccurate Software Development Cost Estimates..* Journal of Systems and Software. V.31, 125-134. 1995.
- [40] Lopez Martin, Cuathemoc et al. *Software development effort estimation using fuzzy logic: a case study*. Proceeding ENC '05 Proceedings of the Sixth Mexican International Conference on Computer Science. 2005.
- [41] McConnell, Steve. *Software Estimation: Demystifying the Black Art*. Best Practices. Microsoft. March 2006.
- [42] Mk II. Function Point Analysis *ISO/IEC 20968:2002* . ISO/IEC 20968:2002 Software engineering - Mk II Function Point Analysis - Counting Practices Manual. 2002.
- [43] Molokken, K., Jorgensen, M., Tanilkan, S.S., H, Gallis, Lien, A.C. and Hove, S.E. *A survey on software estimation in the Norwegian industry*. In Proceedings of the 10th International Symposium on Software Metrics. pp. 208–219. 2004.
- [44] Molokken-Ostvold, K. Haugen, N.C. *Combining Estimates with Planning Poker-An Empirical Study*. ASWEC '07 Proceedings of the 2007 Australian Software Engineering Conference. 2007.
- [45] Nasir M., Ahmad F. *An Empirical Study to Investigate Software Estimation Trend in Organizations Targeting CMMI*. Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science and 1st IEEE/ACIS International Workshop on Component-Based Software Engineering, Software Architecture and Reuse, 2006.
- [46] NESMA *Netherlands Software Metrics Association*. <http://www.nesma.nl/section/home/> (Visitado en 2012).
- [47] NESMA FPA Method *ISO/IEC 24570:2005*. Software engineering - NESMA function size measurement method version 2.1 - Definitions and counting guidelines for the application of Function Point Analysis. 2005
- [48] Norma UNE 71045-1:2000. *Tecnología de la información. Medida del Software. Medida del tamaño funcional..* Parte 1: Definición de conceptos. 2000.
- [49] Páez Anaya, Iván Darío, Anaya Raquel. Travassos H. Guilherme. *Estado Actual de la Estimación de Software en Compañías Colombianas que han Adoptado CMMI*. Experimental Software Engineering Latin American Workshop. ESELAW 2012. Buenos Aires, Argentina. Abril 2012.

- [50] Phan, D. *Information Systems Project Management: an Integrated Resource Planning Perspective Model*. Department of Management and Information Systems. Arizona: Tucson. 1990.
- [51] Planning Poker *Official Website*. <http://www.planningpoker.com/> (Visitado en 2012)
- [52] PMBOK Guide *A Guide to the Project Management Body of Knowledge*. Fourth Edition. Project Management Institute (PMI). 2008.
- [53] R. van Solingen, E. Berghout, *The Goal/Question/Metric Method: a Practical Guide for Quality Improvement of Software Development*. McGraw Hill, London, 1999.
- [54] RCC-TIC *Red Colombiana de Calidad para el Desarrollo en Tecnologías de Información y Comunicación*. <http://www.rcctic.org/> (Visitado en 2012)
- [55] Sauer, C. and Cuthbertson, C. *The State of IT Project Management in the UK 2002-2003*. Templeton College, University of Oxford. 2003.
- [56] Scott, G. *Strategic Planning for High-Tech Product Development*. Technology Analysis & Strategic Management, Vol. 13, No. 3. 2001
- [57] Scrum Methodology *Official Website*. <http://scrummethodology.com/> (Visitado en 2012)
- [58] Shari Lawrence Pfleeger, Rand Inc. *Evaluating Software Technology*. Tutorial at SBES'2002 and Software Engineering Thesis Workshop talk at SBES'2002
- [59] Shaw, M. *Writing Good Software Engineering Research Papers - Minitutorial*. In Proc. of the 25th Intern. Conf. on Software Engineering (ICSE'03), Portland, Oregon, IEEE Computer Society, 2003, pp. 726-736.
- [60] Software Project Cost Estimates *Using COCOMO II Model*. <http://www.codeproject.com/Articles/9266/Software-Project-Cost-Estimates-Using-COCOMO-II-Mo> (Visitado 2012)
- [61] Sprint *Planning Rules and Best Practices*. <http://www.sprintplanning.com/SprintPlanningRules.aspx> (Visitado en 2012)
- [62] Subramanian G. H. , P. C. Pendharkar, M. Wallace. *An empirical study of the effect of complexity, platform, and program type on software development effort of business applications*. Empirical Software Engineering, 2006.
- [63] SWEBOK *The Software Engineering Body of Knowledge..* <http://www.swebok.org/> (Visitado en 2012)

- [64] Travassos G. H. , S. L. Pfleeger, V. R. Basili. *Experimental Software Engineering: an Introduction*. 1st Experimental Software Engineering Latin American Workshop - ESELAW. 2004.
- [65] Travassos, G. H. ; Santos, P. S. M ; Mian, P. ; Dias-Neto, A. C. ; Biolchini, J. *An Environment to Support Large Scale Experimentation in Software Engineering..* In Proc. of XIII IEEE International Conference on Engineering of Complex Computer Systems, Belfast. 2008.
- [66] Travassos, G. H., Kalinowski, M. *iMPS 2010: Desempeño de las empresas que adoptaron el modelo MPS de 2008 a 2010*. Campinas, SP: SOFTEX, 2011.32p. 2010.
- [67] Travassos, G.H. Gurov D. and Amaral, G. *Introdução a Engenharia de Software Experimental*. Relatório Técnico PESC 590/02. COPPE/UFRJ. 2002.
- [68] UKSMA *United Kingdom Software Metrics Association*. <http://www.ukσμα.co.uk/> (Visitado en 2012).
- [69] Wideband Delphi estimation process *Applied Software Project Management*. . O'REILLY. Chapter 3. <http://www.stellman-greene.com/aspm/content/view/23/38/> (Visitado en 2012)
- [70] Wohlin, C. et al. *Experimentation in Software Engineering An Introduction*. International Series in Software Engineering). Kluwer Academic Publishers, USA, 2000.
- [71] Wolter J. Fabrycky. *Análisis del Coste del Ciclo de Vida de los Sistemas*. Editorial ISDEFE. Primera Edición. Madrid, España. 1997.